

**DATA MANAGEMENT SYSTEM****BACKGROUND OF THE INVENTION****1. Field of the Invention**

The present invention relates to a data management system that permits users to view and/or edit a single copy of data under unified relationship criteria. More particularly, the present invention relates to an editorial and/or advertising data management system having enforced relationships between individual components and rules of the components, where the component data is maintained in a single, unified format that can be viewed/edited in overall manner, or by component, or rule.

The present invention has particular utility in the field of editorial and advertisement data management systems that permit users to edit/augment and track publishable stories, by way of insertions (and based on *a priori* knowledge), with various components, while maintaining a unified data structure applied globally to the insertions and components, and will be described in connection with such utility, although other utilities are contemplated.

**2. Description of Related Art****Editorial and Advertising Data Management**

Processing and storage of editorial and advertising data has involved a substantial, and ever increasing amount of complexity. The nature of newspaper publishing has radically altered during the past decade. What was once a relatively straightforward process of producing one or two print editions - all on paper - during a production cycle has swollen to myriad publications, zones, editions and media that must be produced in the same amount of time. One major metropolitan newspaper that used to create two zones in two editions on difficult nights now routinely creates

1 up to 12 zoned versions and five editions in a single evening with concurrent  
2 publications to the Internet.

3 Competition is, of course, the driving force behind this increased diversity.  
4 The market share previously enjoyed by metropolitan publications is now under  
5 assault from a variety of sources, particularly suburban publications, which are often  
6 perceived as better serving markets outside downtown areas, and local broadcast  
7 outlets. Also, newspapers feel compelled to have a strong presence on the Internet  
8 now that it has gained widespread acceptance as an information and commerce  
9 conduit

10 In a separate trend, the weakening of FCC cross-ownership regulations has  
11 resulted in an increase in the number of media companies that own print, broadcast,  
12 cable and on-line publishing media within a single market. With these multiple  
13 outlets, it's understandable that owners would want to reuse content created for one  
14 medium in another. It is becoming increasingly common for a reporter to  
15 simultaneously create content for distribution on the Web, report via broadcast and  
16 create a "traditional" story for the local newspaper. Fundamentally, editors and  
17 reporters are expected to produce a considerably greater amount of content for more  
18 media in the same amount of time it took to previously create a much smaller amount  
19 of content for a single medium.

20 The editorial systems used by these publications have remained relatively  
21 unchanged in the way they manage content since the 1970s and 80s. Editors and  
22 reporters generate individual news stories with virtually no system-maintained  
23 knowledge of a newspaper's zoning and edition requirements. All information  
24 regarding where a story is to be run is maintained either in editors' heads or "cheat  
25 sheets." If a story is to be run in multiple media, editions and/or zones, it is either

1 duplicated or published unmodified. If the story is copied, an additional set of content  
2 management problems arise, particularly when changes must be made in each  
3 autonomous version of a file. Minor, but critical modifications to content often result  
4 in "fishing expeditions" in the database to ensure that all copies of a story are properly  
5 updated. Eventual output of the finished product to archive systems is also  
6 problematic, as libraries are forced to cull through different stories to determine which  
7 ones should be "morgued."

8         For example, referring to Figure 1, a conventional editorial management  
9 system is depicted in block diagram form. In this example, a story 1 may be created  
10 which may comprise a complete publishable story. Naturally, the story 1 contains  
11 content 2, which may include body text, headlines, pictures, etc. The content is  
12 typically associated with the story in traditional, folder/file database storage. Story 1,  
13 with its associated content 2, is used to generate a publication instance 3. Publication  
14 instance 3 may be, for example, a complete story on page 3 of a newspaper (which  
15 may include the above-noted content). If, however, the same (or different) user seeks  
16 to use the same content 2' for the same story 1', but wishes to create another  
17 publishable instance 3' (for example, the same story run in a different zone/edition, or  
18 an entirely different medium (e.g., Internet, etc.)), the user must recreate (e.g. copy)  
19 the content and edit it for the additional publishable instance 3'. In other words,  
20 publishable instances 1 and 1' have no link between each other, and thus, each must  
21 be created as separate, independent (autonomous) versions on the database. Thus, the  
22 same content must be duplicated for each publishable instance, and each copy resides  
23 separately and independently on the database (by way of separate folders, etc.).  
24 Additionally, since no link is established between the publishable instances 1 and 1',  
25 this system can give no information to another user about where other identical

1 content may exist on the database, or whether the content was copied and used for  
2 another publishable instance. Since, as noted above, it is preferable (and, indeed,  
3 required) that the same information (content) is reused for different publishable  
4 instances, this system cannot accomplish this goal, since each instance must be  
5 created separately and independently from all other instances, which can quickly  
6 produce a vast and unmanageable database.

7       Anther example of a conventional editorial management system is shown in  
8 Figure 2. In this example, a master version of a single story 4 (which typically  
9 includes content, as noted above) is stored on a database, and users are permitted to  
10 access the "master" copy, make a subordinate copy, and save any revisions separately  
11 on the database. Thus, for a given story 4, the database of this system may also  
12 include subordinate versions 5, 6 and 7 of the "master", where each version is created  
13 and/or edited independently of any other versions. As shown in the figure, no link is  
14 established between each version, and thus, each version is created and/or edited as a  
15 separate entity from all other versions. Thus, like the example shown in Figure 1,  
16 each version must be maintained separately on the database, since the content must be  
17 created and edited separately from a "master" version.

18       To help resolve some of these story management issues, existing editorial  
19 system usually rely on a highly-serialized form of file management. Publication of  
20 content must occur in a well-defined set of steps or the entire process may collapse.  
21 For example, most publications rely on a system whereby content destined for  
22 publication on the World Wide Web (Internet) is gleaned from listings of stories that  
23 have been previously typeset. The immediacy and potential benefits of simultaneous  
24 (or pre-print) publication of content are lost due to limitations in existing products.

25

1           One attempt to solve the aforementioned shortcomings in conventional  
2 editorial systems can be found in U.S Patent No. 5,181,162.           This patent  
3 discloses an object-oriented document management and production system in which  
4 documents are represented a collections of logical components, or "objects", that may  
5 be combined and physically mapped onto a page-by-page layout. This patent  
6 discloses objects as "content" – text, images, audio, or graphics. These objects may  
7 contain attributes specifying a relationship to other objects or content, or other criteria.  
8 However, this patent does not provide a mechanism by which content can be reused  
9 across multiple mediums, editions, zones or dates, without the need for making a copy  
10 of the content for each instance.

11           Advertising systems have obviously been expected to accommodate the same  
12 zoning, edition and medium challenges faced by newsrooms; it is, of course,  
13 publishers' desire to better target audiences that is pushing much of the need for the  
14 increased workload.

15           The Atex ® Media Solutions Enterprise<sup>TM</sup> Configuration Manager is one  
16 example of an application that allows for the modeling of how a publications products  
17 are scheduled and structured.

18           As a corollary to their modeling capabilities, advertising systems are able to  
19 manage enormously complex advertisement scheduling requirements using simple  
20 user interface techniques that prevent users from scheduling advertising material in  
21 an invalid publication, publication date or physical location within the published  
22 product. Finally, advertising systems have done a better job than their editorial  
23 brethren in monitoring the production status of each of the components that makes up  
24 an advertisement. Whereas an editorial story may consist of only a few text-based  
25 components almost preferably produced by the newspaper staff or a wire service,

1 complex display advertisements may have dozens of components generated by a  
2 variety of sources both internal and external to the newspaper.

3 An additional problem arises because most existing editorial system databases  
4 are capable of storing only textual material. Audio, video, graphics, photographs and  
5 other "multimedia" content must be maintained on separate systems, each with its  
6 own user interface and functionality. Failure of any of these subsystems can result in  
7 fairly spectacular problems with the entire production process. Simultaneous creation  
8 of content for broadcast, Internet and print media is virtually impossible with existing  
9 advertising and editorial products. Additionally, like existing editorial products,  
10 advertising products have not been designed to handle multimedia content (e.g.,  
11 audio/video data, etc.), nor do current advertising products have the ability to define  
12 content to destinations beyond print or Internet, i.e., defining content to appear in  
13 broadcast mediums/media.

14 At the heart of the system is a story management paradigm not contemplated  
15 in the prior art. Rather than thinking of stories as being entities unto themselves  
16 which must be either reused unmodified in multiple editions and/or zones (Figure 1)  
17 or duplicated in order to accommodate modifications specific to a particular medium,  
18 zone or edition (Figure 2), the present invention uses an editorial "insertion" model  
19 that has not been attempted in the editorial and advertising field. A single story is  
20 preferably not duplicated when it is to be run in more than once. Instead, a new  
21 insertion is created. There is no presumption that the publication of one medium is  
22 superior to any other. With the present invention, a story may start life destined for  
23 broadcast, but concurrently be produced in print, broadcast and on-line; the output  
24 medium is irrelevant.

1           At a user's discretion, a new insertion can be directed to "mirror" an existing  
2 one: When a user changes an insertion, the content of another is automatically  
3 updated, eliminating the need to make modifications for each. Alternatively, each  
4 insertion can be independent modifiers of contents, each linked to the same (or  
5 different) content. Since content is defined for a particular destination by way of an  
6 insertion, there is no need to create multiple copies of the same content for each  
7 publishable instance or event, quite unlike prior art advertising and editorial products  
8 known in the art. In other words, the present invention does not manage data using  
9 conventional "file-folder" data structures, rather the insertion can be used to direct a  
10 single copy of content to multiple destinations.

11           The database of the present invention, like the database of the existing Atex  
12 Media Solutions DewarView® editorial product (and some other systems), is capable  
13 of storing a virtually unlimited array of information. However, it may also serve as a  
14 conduit to conventional subsystems, if desired. Textual material, graphics,  
15 photographs, audio and video are accommodated. Indeed, the present invention need  
16 not predict what types of material newspapers will be expected to produce in the years  
17 to come; the system makes no presumptions about the type of material it will be  
18 expected to manage. Additionally, the present invention provides an interface to  
19 permit users to interact with this myriad of data via a single user interface.

20           Importantly, insertions are created based on publication configuration and  
21 scheduling information maintained in the database, and referred to herein as "*a priori*  
22 knowledge". The same rules applied to the creation of advertising content will be  
23 used in the generation of editorial content; it will be impossible to schedule an  
24 editorial story for publication in an invalid publication, nor will it be possible to

1 assign an invalid type of content to a particular publication (as in the case of an audio  
2 snippet being assigned to a print publication).

3 Additionally, the present invention goes a step further, dividing a particular  
4 insertion into components, which generally represent blocks of content (e.g., headline,  
5 body text, photo, photo caption, graphic, audio clip, video clip, etc.). Each of these  
6 components may be edited, paginated and tracked separately from any other  
7 component assigned to an insertion, much the way the individual components of an  
8 advertisement are produced autonomously, then brought together near the end of the  
9 production process.

#### 10 SUMMARY OF THE INVENTION

11 In one aspect, the system of the present invention is designed to improve  
12 existing editorial and advertisement systems by permitting a single copy of content  
13 data to be defined by a plurality of insertions to be destined for a plurality of different  
14 publications, publication mediums, zone, editions, dates, etc, and which can be  
15 updated and/or edited by one or more users. In accordance with the present invention,  
16 a central database stores a unified logical data structure representative of a publishable  
17 story. To accommodate modifications, the central database permits a user (or users)  
18 to create insertions for that data which describes the form that the data will take. Each  
19 insertion can contain one or more components which include content data, e.g.,  
20 headlines, photos, photo caption, logos, advertisement text, etc. Likewise, each  
21 component (i.e., data content) can be associated with several insertions, thereby  
22 "directing" that component to several publications, zones/editions, or into a different  
23 medium, without the need for multiple copies of the same content. Moreover, each  
24 insertion applies a local set of formatting rules to be applied to the components of that  
25 insertion. In addition, the central database tracks each insertion (and each component)



1 created by one or more users, and can globally update any change in one individual  
2 component to all other like-components in other insertions, thereby saving the time  
3 and effort of making multiple copies of data and allowing all users to view/edit any  
4 and all insertions and components. The system also stores all publishable stories,  
5 insertions and components as a single, logical data structure that can be globally  
6 viewed and edited from an insertion, component or story perspective basis, by  
7 associating all components and insertions of a publishable story under a logically  
8 unified data structure.

9 In one embodiment, the present invention provides a data management and  
10 editorial system, comprising a central database for storing at least one publishable  
11 story. The story comprises data content and at least one insertion related to the data  
12 content. The insertion includes one or more attributes related to the data content. At  
13 least one attribute defines a destination for the content. The central database is  
14 adapted to relate the content and insertions of the story in a single, unified data  
15 structure to permit said all the insertions and/or content of said story to be viewed  
16 and/or edited.

17 In another embodiment, the present invention provides a data management and  
18 editorial system for the production and management of publishable events,  
19 comprising: a central database for storing at least one publishable story. The story  
20 comprises data content and at least one insertion related to the data content. The  
21 insertion includes one or more attributes related to the data content wherein at least  
22 one attribute defines a destination for the content. In this embodiment, content is to  
23 appear in multiple insertions. Accordingly, content is defined by separate insertions to  
24 appear in a plurality of separate publishable events, wherein the database links all

1 instances of the content and updates any changes in the content to all instances of the  
2 content.

3 In yet another embodiment, the present invention provides a data management  
4 and editorial system to link content to common pages across multiple publications,  
5 zones, dates and/or editions. In this embodiment, an editorial system is provided for  
6 the production and management of publishable events, comprising: a central database  
7 for storing at least one publishable story. The story comprises data content and a  
8 plurality of insertions related to the content. Each of the insertions including one or  
9 more attributes related to the data content wherein at least one attribute defines a  
10 destination for the content. The central database also for storing common page data  
11 which defines content that is to appear substantially the same on a given page in  
12 multiple destinations. The central database links all the insertions that define a  
13 destination that is part of a common page, and updates any changes in any one linked  
14 insertion with all other linked insertions. Common pages, as defined herein, logically  
15 extend to other non-print media as well.

16 In method form, the present invention provides method of creating a  
17 publishable story, comprising the steps of: creating one or more content items related  
18 to a publishable event; creating at least one insertion for the content items, the  
19 insertion including attribute data including destination data to direct the content to a  
20 specified destination; associating all the content items and said insertions related to  
21 said publishable story to form a single, unified data structure; and storing the  
22 content items, insertions and associations on a database.

23 The present invention also provides a method of creating a publishable story,  
24 comprising the steps of: creating one or more content items related to a publishable  
25 event; creating at least one insertion for the content items, the insertion including

1 attribute data including destination data to direct the content to a specified destination;  
2 creating separate instances of a single copy of the content item and linking all  
3 instances of the single copy of the content item defined by the insertions; updating any  
4 changes in the single copy of the content item to all other linked instances of the  
5 content item; associating all the content items and the insertions related to the  
6 publishable story to form a single, unified data structure; and storing the content  
7 items, insertions and associations on a database.

8 Additionally, the present invention provides a method of creating a publishable  
9 story, comprising the steps of: creating one or more content items related to a  
10 publishable event; creating at least one insertion for the content items, the insertion  
11 including attribute data including destination data to direct the content to a specified  
12 destination; determining if common page rules exist and linking all insertions whose  
13 content appears on common pages. associating all the content items and the insertions  
14 related to the publishable story to form a single, unified data structure; and storing the  
15 content items, insertions and associations on a database.

16 In another aspect, the present invention also provides workflow management  
17 system to configure the workflow of publishable stories, insertions and/or  
18 components. In addition, the system of the present invention permits any user to  
19 obtain the status (e.g., pre-press, finalized, etc.) and view/edit any publishable story,  
20 insertion and/or component. The workflow management system, as provided herein,  
21 is media independent and applies equally to any publishable story in any medium, and  
22 applies routing and constraining procedures to a given publishable story, so that story  
23 can be tracked from its conception to its final disposition.

24 It will be appreciated by those skilled in the art that although the following  
25 Detailed Description, Appendix A and Appendix B will proceed with reference being

1 made to preferred embodiments, the present invention is not intended to be limited to  
2 these preferred embodiments. Other features and advantages of the present invention  
3 will become apparent as the following Detailed Description proceeds, and upon  
4 reference to the Drawings, wherein like numerals depict like parts, and wherein:

5 BRIEF DESCRIPTION OF THE DRAWINGS

6 Figure 1 is a block diagram of an editorial data management system of the  
7 prior art;

8 Figure 2 is a block diagram of another editorial data management system of  
9 the prior art;

10 Figure 3 is a logical relationship diagram of one embodiment of the present  
11 invention;

12 Figure 4 is a story-view logical diagram of the data management system of the  
13 present invention;

14 Figure 5 is an insertion-view logical diagram of the data management system  
15 of the present invention;

16 Figure 6 is a component-view logical diagram of the data management system  
17 of the present invention;

18 Figure 7 is a functional block diagram of the data management system of the  
19 present invention;

20 Figures 8A and 8B depict flowchart diagrams of the preferred creation/editing  
21 of the preferred *a priori* database of the present invention;

22 Figure 9 is a logical relationship diagram of the story, insertions and  
23 components of the present invention where the components remain unlinked;

24 Figure 10 is a logical relationship diagram of the story, insertions and  
25 components of the present invention where the components are linked;

1 Figure 11 is a logical relationship diagram of the story, insertions and  
2 components of the present invention where the insertions are linked; and

3 Figure 12 is a flowchart depicting the preferred creation/editing of stories,  
4 insertions and components of the of the present invention.

5 **Detailed Description of the Invention**

6 While not wishing to be bound by example, the following description will  
7 detail the various components and functionality of the data management system of the  
8 present invention in relation to creating a publishable story. A publishable story is  
9 made up of multiple components in the form of headlines, pictures, text, bylines, etc.  
10 All, or some subset, of the components can be collectively published in various  
11 publications, zones, editions, or other media (e.g., magazines, Internet, newsletters,  
12 broadcast, etc.). Advertisements have the same relationship (i.e., insertion-component  
13 relationship), as do other material published in a newspaper. Collectively and  
14 individually, stories, advertisements, folios, banners, indexes, graphics, rules, etc are  
15 referred to herein as "publishable stories".

16 As used herein, the terms "publishable story" and "story" should be viewed  
17 broadly as media-independent publishable events that can be viewed and/or printed  
18 and/or broadcast simultaneously or independently in a plurality of different mediums.  
19 Accordingly, publishable stories can encompass editorial events (e.g., news stories,  
20 etc.) and/or advertisement events (e.g., classified advertisements, retail  
21 advertisements, etc.) for a given medium. A publishable story can be formatted for  
22 print media and the Internet, while being simultaneously formatted for broadcast  
23 television and radio. A story will also have associated header data fields that can be  
24 user-supplied, or automatically assigned or derived by the system. (Exhibit A) As  
25 used herein, the term "insertion" should be viewed quite broadly as encompassing

1 rules and/or instructions as to the form and destination of a particular component or  
2 set of components. Collectively, an insertion and its components define a  
3 publishable event. For example, an insertion contains attributes (e.g., formatting  
4 instructions, style, font, length, etc.) about the form of a particular component for a  
5 particular medium (e.g., print media, Internet, broadcast, etc.) and/or edition (e.g.,  
6 morning news print edition, etc) and/or geographic zone and/or other criteria. Print  
7 media can include newspapers, publications, magazines, periodicals, directories,  
8 catalogs, newsletters, inserts, etc. Broadcast should be viewed broadly as any  
9 modulated carrier frequency communications channel, including, but not limited to,  
10 cable broadcast communications, free-space broadcast communications (e.g., radio,  
11 UHF, VHF, etc.), satellite communications, digital communications, etc. The  
12 insertion as described herein preferably includes a plurality of data fields where each  
13 data field defines a particular attribute. At its most atomic level, the insertion defines  
14 a destination for content. An insertion will have associated header data fields  
15 (attributes) that can be user-supplied, or automatically assigned or derived by the  
16 system. (Exhibit A). The attributes can be predefined, or may be provided by a user.  
17 Also, as used herein, the term "component" should be viewed broadly as the content  
18 of a particular insertion. Thus, for example, a given insertion can contain a subset of  
19 components, e.g., headlines, editorial text, picture, graphic, caption, audio, video,  
20 advertisement text, logos and/or other data that is computer-readable/transmittable and  
21 /or computer-controllable/reproducible. Likewise, the component includes the content  
22 and an associated header data fields that can be user-supplied, or automatically  
23 assigned or derived by the system. (Exhibit A). Collectively, a complete publication  
24 is a coherent collection of stories, related insertions and, in turn related components.

1           “Application Program” as used herein means that set of computer algorithms  
2     permitting users to create/modify stories, insertion and components. Preferably,  
3     “application program” includes an appropriate program modules to permit users to  
4     create/modify content (components) in a variety of formats, for example, text,  
5     pictures, audio, video, or other publication content. This may include, for example,  
6     pagination software and directory management software that is appropriately adapted  
7     to interface to the system of the present invention to permit users to supply the  
8     appropriate data, as described below. Such software may include proprietary  
9     pagination and editing software tools (which preferably includes page layout and  
10    design software, as is understood in the art) and directory management software, or  
11    may include conventional pagination/editing and/or directory management software  
12    tools. Additionally, the preferred application program includes a “database program”  
13    which permits users to store content information on the database. “Database  
14    program”, in turn, preferably includes an appropriate interface to permit users to query  
15    the database (e.g., using SQL translators), and to define relational data associated with  
16    stories, insertions and components.

17           As the following detailed description proceeds, it is important to note that  
18    components (content) cannot be created for use in a publishable event without being  
19    assigned to an insertion (at some stage of the publication process). (However,  
20    content can exist on the database that has yet to be defined in an insertion). In other  
21    words, the insertion, as set forth herein, provides identity to the content or groups of  
22    content data by logically associating all content under that insertion’s control.  
23    Accordingly, it will be assumed herein that all content, as defined above and herein,  
24    has at least one associated insertion, and that the insertion governs, at the very least,  
25    where that content will appear (destination). One feature of the present invention is

1 the ability for permissive (or restrictive) rules to be placed on content, via attributes in  
2 insertions, so that the placement or destination of content cannot conflict with preset  
3 pagination and/or destination criteria.

4 Referring to FIG. 3, the relationship between publishable stories 10, insertions  
5 12 and components 14 is shown. As noted above, a component 14 is a subset of an  
6 insertion, and preferably comprises text, photos, video, audio, etc, i.e., content. An  
7 insertion 12 is defined by a user (or be partially or fully defined *a priori*, as discussed  
8 below), and operates as an instruction set for a given subset of components. Thus,  
9 for example, an insertion defines the particular form (e.g., format) that the  
10 components 14 will take for a given publication medium/zone/edition. In the present  
11 invention, the association between content, insertions and stories includes three  
12 preferable management structures: 1) the content remains independent across  
13 insertions, thus, each instance of the content can be edited separately, to  
14 accommodate, for example, physical size limitations of a particular medium; 2) the  
15 content is linked between insertions and thus, a given component can appear in  
16 multiple insertions, or, alternatively, between insertions, the form (e.g., length, font  
17 format, style, etc.) of the component can vary. Thus, while the content can remain the  
18 same, the form may vary between insertions. This does not mean that multiple copies  
19 of the content are required, just that multiple insertions define the destination of that  
20 content that is to appear substantially unchanged in each of those insertions. 3)  
21 Common pages are defined in which all insertions and content one page is to appear  
22 identically on another page in another medium, zone, data, or edition. As described  
23 more fully below, all insertions (and, accordingly, all components of each insertion)  
24 are stored by a central database in a single, unified data structure that can be accessed,  
25 edited and tracked by a plurality of users (e.g., editors). Of course, each of the above-



1 mentioned management structures can be combined for any given story and/or  
2 publishable event.

3 FIG. 4 is a logical diagram of one embodiment of the data structure of the  
4 present invention. By way of example, suppose that a certain publishable story 10 is  
5 being created on a "warehouse fire". As shown in the Figure, a user can create an  
6 insertion 12a for the "warehouse fire" for a specified date, zone (e.g., *Boston Globe*,  
7 *NorthZone*), edition (e.g., morning edition news print), and location (e.g., page 3).  
8 That insertion can contain components 14a-14d specific to that insertion 12a. The  
9 insertion 12a contains instructions, via editing software, as to the appropriate format  
10 and style of each of the components 14a-14d, for that zone and/or edition.  
11 Component 14a is depicted as "headline 1" for that particular insertion, and is  
12 formatted accordingly, via insertion 12a, for the particular zone/edition. Component  
13 14b represents the text of the warehouse fire, as created by, e.g., a reporter. Insertion  
14 12a will dictate the format and style of component 14b for the particular zone/edition.  
15 Likewise, components 14c and 14d are under similar constraints, as dictated by  
16 insertion 12a. Insertion 12a' represent the same publishable story, but as it appears  
17 and is formatted for page 8 of the same medium as insertion 12a. Since insertion 12a'  
18 appears at a different physical location compared to insertion 12a, it contains a  
19 different subset of components 14a'-14c'. In this case, components 14a'-14c' are  
20 constrained by insertion 12a' in a similar manner as described above. In addition, a  
21 new insertion 12b can be created, by the same or different user, for the publishable  
22 story 10 that will be displayed on the Internet. In this case, insertion 12b dictates the  
23 format of components 14e-14h for the Internet medium. Thus, for example, a video  
24 clip 14g may be included as a component of the warehouse fire story on the Internet.  
25 Of course, similar components, e.g., 14a, 14a' and 14e, can be exactly the same, or

1 different in both content and form. The data format as shown in FIG. 4 is taken from  
2 the perspective of the publishable story 10 (i.e., "story view"), and displays all  
3 information for that story in single logical data structure, i.e., the collection of  
4 insertions and components under that story, in a single logical data structure.

5       Figures 5 and 6 are logical diagrams of other embodiments of the data  
6 structure of the present invention. The data structures depicted in Figures 5 and 6 are  
7 from the viewpoint of the insertion and component, respectively. It should be noted at  
8 the outset that the data structures of Figures 4-6 are not mutually exclusive, but rather,  
9 are different ways the data management system of the present invention can track  
10 publishable stories, insertions and components. As shown in FIG. 5, the present  
11 invention permits a user to view/edit any and all insertions at a particular physical  
12 location of the medium. Thus, for example, all insertions (and, all components  
13 thereof) present on page 3 of a newspaper can be displayed and edited. Similarly,  
14 from a component view (FIG. 6), all components for any given insertion or  
15 publishable story can be displayed and/or edited.

16       As will be appreciated, an editorial story (story) is only one example of the  
17 data management system of the present invention. Most media publish stories in the  
18 form of advertisements as well. In other words, a given media has both an editorial  
19 component and advertisement component. Advertisements, for a given medium,  
20 come in various forms, e.g., printed, audio, video, graphical, etc, or some combination  
21 thereof. Advertisements can be further broken down into classified, retail, inserts and  
22 other advertisements. The data management system of present invention, as  
23 described above in reference to a news story, applies equally to advertisements. Thus,  
24 for example a given advertisement can contain certain components (e.g., text, banner,  
25 graphic, logos, etc.) specific for a given medium, and can be defined (i.e., constrained)

1 for that medium by way of an insertion. The same advertisement may also comprise a  
2 different set of components (e.g., video, audio, etc.) for a different medium, and be  
3 defined for that medium by a separate insertion. In that regard, the logical structure of  
4 Figures 2-4 (i.e., insertion-component relationship), and indeed, the relationship  
5 between publishable story, insertion and component of Figure 3, extends to  
6 advertisements as well. For example, the publishable story 10 could represent a retail  
7 advertisement that will be run in a variety of newsprint media (including, for example,  
8 newspapers, magazines, periodicals, newsletters, professional association journals,  
9 etc.) and/or in the Internet. Accordingly, insertion 12a is defined as the local set of  
10 rules as to the form of components 14a-14d, as defined for a particular location of a  
11 particular medium. Of course, that same advertisement could be run on the Internet  
12 (by way of insertion 12b) and could comprise, e.g., video clips, audio clips, text,  
13 logos, and graphics as components thereof.

14       The central database may comprise a stand-alone system, or can comprise one  
15 or more distributed computer program processes running on one or more networked  
16 personal (e.g., Apple McIntosh™ or Intel Pentium™-based) and/or mainframe  
17 computers and include such additional computer, mass storage, and communications  
18 hardware and software as is appropriate to enable performance of their stated  
19 functions. For example, the computers may run the Microsoft Windows™, Windows  
20 NT™, or DOS™ operating systems, and may include database programs such as  
21 Oracle™, or other proprietary and or off-the-shelf database programs known in the art.  
22 Alternatively, the various functional components of the system present invention may  
23 be constructed solely out of custom, dedicated electronic hardware and software,  
24 without departing from the present invention. Further alternatively, rather than use the

1 aforesaid types of computers, the system of the present invention may instead utilize  
2 SUN™ and/or RISC-based workstations running Solaris™ and/or Unix™ operating  
3 systems, or an IBM RS/6000™ running AIX™ operating system. Additionally, each  
4 user can track the status of and perform the various edits to the publishable stories  
5 either locally, or remotely over a network connection. To that end, the system can be  
6 adapted to permit users to download data, work offline to view/edit, and upload the  
7 data at a later time. The network may comprise a TCP/IP-based wide area network  
8 (e.g., an Internet-type of computer network). Of course, the present invention should  
9 not be viewed as being limited to the above-described types of  
10 computer/communications hardware, operating systems, and program processes, but  
11 rather, should be viewed expansively, as encompassing all types of  
12 computer/communications hardware, operating systems, and program processes so  
13 long as the stated functionality for the present invention may be carried out by same.  
14 It should also be noted that the system of the present invention can accommodate  
15 various software text, audio, video and other computer data editing tools known in the  
16 art. To that end, the system of the present invention can be appropriately modified  
17 with a dedicated software editing tool, or other editing packages known in the art.

18 As will be described more fully below, the present invention includes database  
19 structure and management features and associated functionality. For example the  
20 database structures described hereinafter may comprise "relational" database  
21 technology is modeled such that all data is organized as though it is formatted into  
22 tables, with the table columns representing the table's fields or domains and the table  
23 rows representing the values of the table's fields or domains. Data is logically  
24 organized as tables but is not necessarily physically stored as such. The relational  
25 database user does not need to know how the database is physically constructed and

1 can access and update data via a language interface or "structured query language"  
2 (SQL). The relational model described herein between stories, insertions and  
3 components assumes a certain stability in the number of columns of data associated  
4 with a table and that usually, data is present in most, if not all, fields within the table.  
5 However, those skilled in the art will recognize that other database structures and  
6 interfaces can be used without departing from the scope of the present invention.

7 Figure 7 is a functional block diagram of the preferred embodiment of the data  
8 management system 30 of the present invention. Essentially, the system 30 of the  
9 present invention includes an administration system 32, an *a priori* database 38, a  
10 central database 34, an output manager 36, and an application program / database  
11 program 44. It should be noted that each of the functional components depicted in  
12 Figure 7 may include separate modular systems, or may be part of a unified database  
13 system. Each of these functional components is discussed more fully below.

14 Administration system 32, which includes functionality to establish the *a*  
15 *priori* database 38, maintain the central database 34 and establish global rules applied  
16 to users 46A, 46b, 46C...46N, preferably includes a configuration manager 40.  
17 Configuration manager 40 includes appropriate database functionality to permit a user  
18 (in this case, a system administrator, for example) to insert information (i.e., via data  
19 fields) related to the particular operating environment of the system. For example, a  
20 given publication will have *a priori* knowledge as to the scope of that publications'  
21 coverage, the different media in which publication events could occur, the different  
22 zones that publication may publish, times, dates, sections, editions, etc., and other  
23 knowledge specific to that publication. Additionally, the configuration manager 40  
24 can include restriction and /or limiting data that can be placed globally on any data  
25 field within an insertion. For example, restriction data can be placed on a given

1 publication date or edition such that no content (components) can be placed on page  
2 30 of that data or edition (for example, either because page 30 does not exist for that  
3 publication, or that other content is pre-reserved for that page). This list is only  
4 exemplary information, and other configuration data can be added to the system  
5 without departing from the scope of the present invention. Additionally, common  
6 page data can be defined for certain insertions (or, more particularly, attributes of  
7 certain insertions) to define common pages that may exist across several editions  
8 and/or zones of a given publication. For example, it is possible with the present  
9 invention to predefine content and insertions for a given page in a given publication  
10 zone/edition/medium to mirror another page in another zone/edition/medium.

11 As part of the functionality associated with configuration manager, user's can  
12 define details related to preexisting content, shape, size and other dimensional  
13 characteristics specific to a particular publication and/or publication medium. For  
14 example, users are permitted to store recurrent data related to the overall dimensions  
15 of a given publication, specific dimension of a given page of a publication, etc.  
16 Additionally, certain content (components) recur from publication to publication,  
17 across zones, across editions, and across mediums. For example, logos, page number  
18 identifiers, templates, etc. may be common to all publications. Common page data  
19 can be defined for certain insertions (or, more particularly, attributes of certain  
20 insertions) to define common pages that may exist across several publications,  
21 editions, mediums and/or zones. For example, it is possible with the present invention  
22 to predefine content and insertions for a given page in a given publication  
23 zone/edition/medium to mirror another page in another zone/edition/medium.  
24 Common page data is one example of preset insertions that can be established *a priori*  
25 using configuration manager 40.

1 Data generated by the configuration manager 40 is stored in the *a priori*  
2 database 38. As will be described below, the structure of the *a priori* database is  
3 preferably in the same (or equivalent) format as the insertion structure, since the  
4 insertion is checked against the database 38 for conflicting information. In other  
5 words, in the preferred system both the central database and the *a priori* database 38  
6 are constructed using a "common currency" database language and structure.  
7 Alternatively, an appropriate database conversion system can be provided (not shown)  
8 to permit the system 30 to read and query both databases 38 and 34. In any event,  
9 administration system 32 preferably include an appropriate user-interface to permit  
10 entry of *a priori* data into the database 38. Of course, the interface may include, for  
11 example, prompt-type in which a user is prompted with a series of choices regarding  
12 the above-described particulars of a given environment, or field-type in which a user  
13 enters information into preset or user-definable data fields. Additionally, especially  
14 with respect to preexisting content, the system 32 may also include data filters and/or  
15 translators (not shown) to convert preexisting content into a preferred format. To that  
16 end, additional databases may be present which store preexisting data, which may be  
17 directly transferred to database 38 or logically stored in database 38 (using, e.g.,  
18 pointer data).

19 In addition to, or as part of, configuration manager 40, a module is also  
20 preferably provided which permits global (i.e., system-wide) control over user and  
21 content rights. Thus, for example, configuration manager 40 (or some subset module)  
22 can configure the system 30 to include a database of permitted user's of the system,  
23 and for those user's, provide what kind of access is granted (e.g., full edit rights,  
24 partial edit rights, view only, etc.) for specified content, or these preset rights may be  
25 defined globally for all content within the system. This may further include

1 appropriate security measures (e.g., password protection, etc.) that can be applied on a  
2 per-user basis, and/or a per-content basis.

3 Referring to Figures 8A and 8B, flowcharts are depicted which detail the  
4 construction of the *a priori* database 38, using the configuration manager 40 to  
5 configure the system and define particular characteristics of a publication. It should  
6 be noted that the process steps indicated in Figures 8A and 8B are exemplary  
7 processes that may be associated with configuration manager 40, and are not intended  
8 to be an exhaustive list of their respective stated functionality. It should also be noted  
9 that unless otherwise specified, the sequence of steps listed in Figures 8A and 8B need  
10 not be performed in the listed sequence, but may be performed in any sequence to  
11 accomplish the stated functionality. Also, for clarity, the system components  
12 numerically referenced in Figure 7 are omitted.

13 An administrator (e.g., system administrator) opens configuration manager 50  
14 to begin defining the particular operating environment of the system 30. The  
15 administrator creates and/or modifies a list of users 52 permitted to access/view/edit  
16 data on the system. The administrator can create/modify groups 54 by logically  
17 linking groups of users that share, for example, common roles in the publication  
18 process. Thus, for example, groups (of users) can be defined by department, title, etc.  
19 For each user and/or groups of users, parameters can be assigned 58. For example,  
20 access rights, password information, etc. can be assigned to users or groups of users.  
21 The administrator creates/modifies queues 56. A queue is that data related to  
22 workflow of a given story (including insertion and components), from conception to  
23 final publication. Thus, a queue can be assigned parameters 60 which may include,  
24 for example, the number of hours a story (including that story's insertions and  
25 components) lies "dormant" (i.e., period of time a story remains unaccessed) in the



1 database before moving that story to final output or deleted, etc. Additionally, user  
2 access to any particular queue or queues 62 can defined/modified. The administrator  
3 can define default parameters 64 such as: scope of a publications' coverage, the  
4 different zones that publication may publish in, times, dates, sections, editions,  
5 different media of publication (e.g., newspaper, Internet, periodical, broadcast, etc.),  
6 and other knowledge specific to that publication. The administrator also preferably  
7 creates/modifies restriction data 66 specific for a given publication, medium, time,  
8 zone, edition, etc. This data can include, for example, absolute page data, page count,  
9 etc. This data is stored in the *a priori* database 68, and is used in the publication  
10 production process, as described herein.

11       Additionally, the administrator or other user can open the configuration  
12 manager 70 (Figure 8B), to define data related to preexisting content and/or shape,  
13 size and other dimensional and layout characteristics specific to a particular  
14 publication and/or publication medium. For example, the overall layout of a  
15 publication can be defined 72, which may include inputting data related to specific  
16 pages. Additionally, size, shape and other dimensional or layout characteristics  
17 related to components (content) can be defined/modified. Pre-existing content can be  
18 assigned and/or modified 74. This type of content, for example, may include  
19 preferred template arrangements for a given page, etc. In most publications, there is  
20 some content that appears constantly, for example, day-to-day in all editions and  
21 zones. Thus, an administrator can define this type of recurrent data 76 (content) and  
22 can further define, for example, that contents' placement on a particular page. As  
23 noted above, common page data can be defined, in which a plurality of preset  
24 insertions are defined 80. Of course, content related to those insertions can also be  
25 assigned to a page, which may be linked to other pages in common. Likewise, this

1 data is stored in the *a priori* database 78, and is used in the publication production  
2 process, as described herein. It should be noted that recurrent data, or indeed any  
3 content or rule structure provided by the present invention, need not reside on the  
4 database, but instead may be associated with the database via, for example, pointer  
5 data.

6 Referring again to Figure 7, the central database 34 stores the story 10,  
7 insertion 12 and component 14 data. Application program and database program 44  
8 permit users 46A, 46B, 46C...46N to access the database 34 and create/edit (if  
9 permitted) components, insertions and stories. Application program 44 may include  
10 pagination editing tools, management tools, database translating tools, and other tools  
11 as may be necessary. Such tools (programs) can be custom-made or off-the-shelf  
12 programs for editing, paginating and database file management, and other purposes.  
13 Output manager 36 is responsible for accumulating those stories that are common to a  
14 particular publication, zone, medium or edition and logically grouping those common  
15 story destined for a particular output. For example, output manager 36 will query the  
16 database 34 for all insertions that contain a data field for a particular edition in a  
17 particular zone of a newspaper in a particular medium. Output manager 36 includes  
18 appropriate systems to manage and/or translate and/or format a given collection of  
19 stories, insertion and components that are destined for a particular medium. For  
20 example, output manager 36 preferably includes a web-based publishing content  
21 manager to appropriately format those stories, insertion and components destined for  
22 the web. Other examples may include broadcast systems to manage those stories,  
23 insertion and components destined for broadcast (e.g., television, radio). The  
24 pagination rules supplied by the *a priori* database 38 and those created in the

1 insertions will direct the final output of all the appropriate insertions for that  
2 publication.

3 Archival system 42 is provided to permit storage of any of the insertions,  
4 components (content) and/or stories herein defined for future access and retrieval.  
5 Thus, for example, a user may wish to archive a given publication event and define an  
6 insertion that directs the destination for the content of that event at some future time.  
7 Archival system 42 can also include such mass storage devices to store audio and/or  
8 video data (content) that can be defined by an insertion to be destined for broadcast or  
9 Internet mediums. Instead of having to recreate the content, the user can indicate in  
10 the insertion that the content exists as an archive in the archival system. Archival  
11 system preferably includes appropriately adapted mass-storage devices to  
12 accommodate the various type of content contemplated herein. Data from sources not  
13 shown in Figure 7 can also be stored in archival system, for example, external data,  
14 video, graphics, audio, and/or other data.

15 Additionally, communications manager 48 is provided in the preferred system  
16 30. Communications manager 48 essentially includes appropriate hardware and  
17 software to permit external wire service content data (e.g., AP/UPI) to be  
18 automatically translated into an appropriate format and assigned an insertion. In this  
19 instance, the insertion is preferably generated automatically by communications  
20 manager based upon data gathered from *a priori* database 38 (e.g., layout, style, etc.),  
21 and/or internal data associated with the content. Once the insertion(s) and  
22 component(s) are created and stored on the database 34, a user may edit or modify  
23 both, as if both were originally supplied by a user. As noted above, content cannot be  
24 published by the system unless it is associated with an insertion. Thus, insertion,  
25 content and story attributes in the preferred system 30 can be supplied by the

1 application program 44 (which provides a mixture of preset and user-specified  
2 insertion data) and the communications manager 48, which automatically assigns  
3 attributes based on data from the *a priori* database. This automatic assignment may  
4 contain one or more data fields (attributes) that are assigned "TO BE  
5 DETERMINED" status, which may be supplied by a user of the applications program  
6 at a future time.

7       Figures 9, 10 and 11 depict preferred embodiments of the logical association  
8 of stories, insertions and components when content differs between insertions, when  
9 content is linked between insertions, and when insertions and content are identical to  
10 multiple pages (herein referred to as "common pages"), respectively. Referring to  
11 Figure 9, a story is created under which insertions and content are provided and  
12 defined. In this example, the content BT1 (body text 1), BT2 and BT3 all differ from  
13 one another (e.g., each is not an exact content copy of the others) for insertion into a  
14 publication medium, via Insertion 1, 2 and 3, respectively. Since it is known that the  
15 content will differ between insertions (and hence, between each destined medium), the  
16 system permits existing content to be edited for another publication, zone, edition  
17 and/or medium, where each new instance of content is defined by another insertion.  
18 Likewise H1, H2, H3 (headlines) and AUDIO (audio clip) differ from each other, and  
19 are likewise unlinked. In Figure 10, the content BT1 and BT2 are identical.  
20 Accordingly, the content is linked together in the database. Any change in BT1 will  
21 automatically generate a like change in BT2. In this example, the link can be between  
22 content only, or between content and/or shape, and/or style, etc., where a change in  
23 one generates a like change in the other. Of course, multiple content can be linked  
24 across multiple insertions. Additionally, not all of the content in a given insertion  
25 need be linked to other content in another insertion. For example, although in this

example BT1=BT2, H1 need not necessarily equal H2 (although, of course, this linkage among all content in an insertion could exist).

In Figure 11, common pages are established in which certain content and insertions of a page are to be mirrored on another page in another medium. In this case, the insertions are linked. Common pages can be defined *a priori* (using the *a priori* database construction processes, described above), or defined at any stage of the process. Common pages, as described herein, includes those media events for which all of the content and insertions are to be duplicated in another media event. One obvious example is defining a common page to exist across multiple publications, where the page will appear identical. Common pages applies equally to non-print mediums, for example, defining common pages for unique web sites, defining common pages for independent broadcast events, etc. Thus, common pages is not restricted to a printed page publications, but rather, for that set of components and insertions that are to be defined to be presented identically (or substantially identically) in any publication medium. Moreover, not all of the insertions for a given publishable event or story need be linked to the other insertions for that event or story. Linked changes in Figures 10 and 11 may be made not only to text but to other components (not all pictured).

Figure 12 depicts the overall flow and decision flowchart 100 for the creation and/or editing of stories, insertions and components that are stored on the database 34, in accordance with the above described embodiments of Figure 9, 10 and 11. Appendix A, at the end of the Detailed Description, provides preferred component, insertion and story header (attribute) data. The data fields listed in Appendix A are provided as examples of the type of data that can be associated with the component, insertion and story, and are not intended to be an exhaustive, exclusive list. Those

1 skilled in the art will recognize that certain data fields (attributes) for the components,  
2 insertions and stories may be user-supplied (via applications program, discussed  
3 above) or predefined (using, for example configuration manager, discussed above ).  
4 Additionally, it will be noted that certain attributes are preferably restricted (locked),  
5 and/or conditionally restricted from a user's perspective, so that a user is not permitted  
6 to edit (change) these attributes. Reference will be made in the following detailed  
7 description to certain attributes listed in Appendix A, but it should be recognized that  
8 similar processes described in Figures 12 may be implemented to define and/or edit a  
9 particular attribute. It should be noted that the process and decision steps described in  
10 Figure 12 do not necessarily need to take place in the order noted, rather Figure 12 is  
11 intended to depict the overall flow of the creation of insertions, stories and  
12 components at any point in the production process. A story is defined, or a pre-  
13 existing story is opened (if permitted) 102. The appropriate header data (Appendix A)  
14 is supplied for the story 104. For that story, content is created or modified (if  
15 permitted) 104, and a header for that content is created/modified 106. Once content is  
16 created, an insertion is defined for that content 110. As noted above, the insertion  
17 contains, at a minimum, data directing the output medium of the content.  
18 Accordingly, the destination (output medium) of that content is defined 112 for that  
19 insertion. Likewise, additional header data is preferably created/modified 114. The  
20 system looks at the header data in the story, insertion and component to determine if a  
21 preset (*a priori*) rule is violated 116. This can include, comparing the user supplied  
22 or automatically generated header data with preset header data to flag a conflict, or  
23 this may include a set of rules about what content can appear in a given medium. If a  
24 conflict exists, the system prompts the user 118 to correct the appropriate entry. The  
25 system queries the database to determine if the content is (or, is not) to be reused in

1 another insertion 120. If not, the content is not linked to other content on the  
2 database, or future insertions to be created 122. (Figure 9). If yes, the content is  
3 linked. 124. (Figure 10). This is performed, for example, by matching header  
4 information in the content with content already on the system, or by a user defining  
5 that the content is to be reused by another insertion. An additional insertion is created  
6 for each instance (destination) of the linked content 126. Since the content is linked,  
7 the system will determine if this is new content, or pre-existing content that is being  
8 modified 128. If the content is pre-existing, the system will update all linked content  
9 with the modifications 130. It is important to note that linked content is not multiple  
10 copies of the same data, but rather, a single copy of data that is logically linked to  
11 different insertions, to be reused in those insertions. The system will also inquire as to  
12 whether common pages have been defined 132. The common page rules are  
13 preferably defined in the *a priori* database, but may also be defined at any stage of the  
14 publication process and stored in the central database. If yes, the system will create  
15 multiple insertions for the content, as indicated by rules defined by the common pages  
16 134. Essentially, these rules indicate that a page is to be mirrored (i.e., exact replicas  
17 of the content and insertions) in two or more publications, zones, editions and/or  
18 mediums. This data is stored on the central database 136.

19 Those skilled in the art will recognize that certain criteria, apart from the  
20 story/insertion/component model described herein, need to be defined in the  
21 publication production process. For example, it is often desirable to define page data  
22 which permits a given page on a particular media to be defined in terms of size,  
23 format, date, edition, zone, slug (shorthand name), and/or other criteria. Additionally,  
24 certain definitions may apply specifically to print media, for example, Double-Truck  
25 To field which contains the identification of a double-truck partner page (this

1 identification should be translated to physical and logical pages depending on how a  
2 user is referencing the component or page), Double-Truck Side field which indicates  
3 whether the page is the left or right side of a double-truck pair (the attributes of this  
4 field are assigned at the time the page is determined to be part of a double-truck),  
5 Logical Section field which displays the logical section in which the component is  
6 intended to be published (in addition to the logical section values specified in the  
7 Configuration Manager, the Logical Section field may be set to TBD — “To Be  
8 Determined.” The Logical Section field is preferably protected; Logical Section  
9 information for a component may be changed either by altering the logical section  
10 field for the insertion to which it belongs, by “dragging and dropping” the component  
11 into another insertion or by invoking manual commands), Physical Section field  
12 displays the physical section in which the component is intended to be published In  
13 addition to the physical sections specified in the Configuration Manager, the Physical  
14 Section field may be set to TBD — “To Be Determined.” The Physical Section field is  
15 preferably protected; Physical Section information for a component may be changed  
16 either by altering the physical section field for the insertion to which it belongs, by  
17 “dragging and dropping” the component into another insertion or by invoking manual  
18 commands), Double-Truck To field which contains the identification of a double-  
19 truck partner page (this identification should be translated to physical or logical page,  
20 depending on how a user is referencing the component or page. The Double-Truck To  
21 field is protected; it is preferably filled in by the system and preferably cannot be  
22 modified by users), Logical Section field which displays the logical section to which  
23 the insertion has been assigned (information in this field may be derived by default  
24 publication sizing information, gleaned from data maintained by the Configuration  
25 Manager, will be displayed. Modifications to the logical section field of the insertion



1 necessarily modify the Logical Section fields of all components it contains. The  
2 Logical Section field will be represented in the insertion header as a combo box with  
3 an associated editable entry field. The user will either type the name of a valid logical  
4 section or may select one from the combo box. The Logical Section field is linked to  
5 the Publication, Publication Date, Zone, Edition, Logical Page, Physical Section and  
6 Absolute Page fields in that a modification to the Logical Section field may result in  
7 the entries in the other fields to change to reflect valid  
8 publication information. In addition to the logical sections specified in the  
9 Configuration Manager, the Logical Section field may be set to TBD — “To Be  
10 Determined.” The Logical Section field may be modified via the insertion header, by  
11 altering a displayed field in the system or by invoking manual commands), Physical  
12 Section field which displays the physical section to which the insertion has been  
13 assigned (Modifications to the Physical Section field of the insertion necessarily  
14 modify the Physical Section field of the components it contains. The Physical Section  
15 field is linked to the Publication, Publication Date, Zone, Edition, Logical Page,  
16 Logical Section and Absolute Page fields in that a modification to the Physical  
17 Section field may result in the entries in the other fields to change to reflect valid  
18 publication information. In addition to the physical sections specified in the  
19 Configuration Manager, the Physical Section field may be set to TBD — “To Be  
20 Determined.” The Physical Section field is preferably protected; Physical Section  
21 information for an insertion may be changed by invoking manual commands), and or  
22 other page criteria. Essentially, page definitional data is where insertions are booked  
23 against, i.e., the manifestation of the insertions. Of course, those skilled in the art will  
24 recognize that additional and/or different parameters may be used to define a “page”,  
25 as used herein.

1           Additionally, insertions layout data can be defined which can include global  
2   and/or local commands by which insertions are fundamentally defined. Insertion  
3   layout data includes the geometric boundaries for the content, and the relationship  
4   between the boundaries of content. Insertion layout data can include, for example, Fit  
5   Indication field used to indicate that the component fits in the available space  
6   (acceptable entries include "Short," "Fits" and "Overset." The Fit Indication field is  
7   protected; it is preferably filled in by the system and cannot be modified by users.  
8   Though the user will see running information indicating whether the component fits  
9   the assigned shape as he or she modifies the content, the Fit Indication field will be  
10   updated only when the user stores the associated component, not each time the story is  
11   edited. The Fit Indication fields of an insertion's components dictate how the Fit  
12   Indication field of the insertion will be set. If any component in an insertion has a fit  
13   indication other than "Fits," the Fit Indication field of the insertion will be set to  
14   "Components Don't Yet Fit." When all components of an insertion fit, the insertion's  
15   Fit Indication field will be set to "All Components Fit."), Shape field specifies the  
16   geometric properties to use when composing the component (initially, the Shape field  
17   contains a default specification which is tuned for the name of the component and the  
18   publication to which it has been assigned. Until the component is assigned a  
19   pagination status of placed, the Shape field is unprotected. Authorized users can  
20   change its contents by selecting from available shapes, displayed when the user pulls  
21   down a combo box associated with the field. Shapes can be a complex list of ordered  
22   polygons or a simple rectangle. The value inserted into the Shape field may include,  
23   for example, a named shape applied by the user, a shape name derived from the  
24   component type and the layout applied to the insertion, a default value inserted by the  
25   system, set to "drawn," which indicates the shape comes from a container drawn in.),

1 Style Set field specifies the composition properties to use when composing a  
2 component (initially, the Style field contains a default specification tuned for the name  
3 of the component and the publication to which it has been assigned. Until the  
4 component is assigned a pagination status of placed, the Style field is unprotected.  
5 Authorized users can change its contents by selecting from available styles, displayed  
6 when the user pulls down a combo box associated with the field.), X Position field  
7 represents the position on the X axis of the upper left-hand corner of the component  
8 (for the insertion, it is the upper left-hand corner of the overall bounding box. This  
9 field is protected if the component or insertion is locked by the pagination editor or  
10 has a status of Placed or higher. When the story is locked by the pagination editor,  
11 only the session holding the lock may change the position. When the status is greater  
12 than Placed, the position may only be changed by the pagination editor), Y Position  
13 field represents the position on the Y axis of the upper left-hand corner of the  
14 component (for the insertion, it is the upper left-hand corner of the overall bounding  
15 box. This field is protected if the component or insertion is locked by the pagination  
16 editor or has a status of Placed or higher. When the story is locked by the pagination  
17 editor, only the session holding the lock may change the position. When the status is  
18 greater than Placed, the position may only be changed by the pagination editor), H&J  
19 (Hyphenation and Justification) Length. This field displays the composed depth of the  
20 component. The H&J field of newly-created user-generated stories is blank until the  
21 story is H&Jd and stored. The H&J length of existing material, though updated  
22 dynamically as the user modifies the content, is not updated in the H&J Length header  
23 field until the user stores the story. The unit of measurement used to display the  
24 composed depth is specified in the user's user pro-file. The H&J field is conditionally

1 protected; it cannot be updated while the associated text is open for edit. The contents  
2 of this field are preferably not modified by the user.  
3 . Of course, those skilled in the art will recognize that additional and/or different  
4 parameters may be used to define a "insertion layout", as used herein.

5 In another aspect of the present invention, a workflow management system is  
6 provided to aid in the tracking and routing of publishable stories, insertions and  
7 components from conception (i.e., creation) to disposition (e.g., publication, archival,  
8 billing, cancellation, etc.) in either an automatic or user-defined manner. Preferably,  
9 the workflow management system applies routing and constraining instructions for a  
10 given publishable story, insertion and/or component. Workflow management is  
11 preferably included as part of the "applications program" discussed herein.

12 Accordingly, the workflow management system of the present invention permits users  
13 to create a set of procedures, that can include adding steps for a given procedure so  
14 that a given publishable story can be routed and tracked according to a pre-defined set  
15 of rules that are user-configurable. Moreover, the status of any publishable story can  
16 also be rule-based and configurable by site according to their specific workflow needs  
17 and is continually updated and can be viewed/edited by one or more users. Routing  
18 decisions for a given publishable story can be based on sequencing, i.e., steps  
19 performed in a certain order, or based on other criteria, e.g., the evaluation of the  
20 attributes of the story or its components. In addition, steps can be linked together to  
21 form a sequence order, i.e., grouped together logically to form procedures. Thus, for  
22 example, a group can be constrained such that the steps are executed at once in a  
23 parallel-logical manner, or, the overall sequence of steps can be constrained such that  
24 all steps are executed in series. Routing decisions can include the parameters of  
25 where the publishable story is to be sent (e.g., physical location, such as the queue

1 where the next person is to work on it), for example, a sports wire story coming in  
2 could be examined and the system determine that it should be sent directly to one of  
3 the queues of the sports desk. To that end, the workflow management system of the  
4 present invention can also include appropriate hardware/software to route publishable  
5 stories (using, for example, output manager, discussed above) between separate users,  
6 via, for example, the Internet or other network, or to different users via a set of queues  
7 that the users would have explicit access to. Routing decisions can also include  
8 logical instructions defining the overall flow from creation to finalization. Preferably,  
9 the workflow management system is comprised of a graphical user interface to permit  
10 users graphically create and define the procedures and steps and to link together steps  
11 in a group for a given publishable story. Also preferably, the workflow management  
12 system comprises Visual Basic<sup>TM</sup> for Applications as the scripting tool for defining  
13 the workflow process as described above, and to provide other functional components,  
14 for example, notification that a story is ready for the next step or that it is running  
15 behind the scheduled deadline for insertion into the publication.

16 For example, an advertisement (a publishable story) typically has several steps  
17 that must be tracked from creation to disposition. The present invention permits the  
18 creation of a set of procedures for defining the sequence of steps, and apply routing  
19 decisions for any given number of steps. An advertisement can include the steps of  
20 creation, editing, publication and billing (not necessarily in that order). In addition, a  
21 credit check for a given client might also be run before the advertisement is finalized  
22 and/or published. The present invention permits a user to define rules associated with  
23 these steps. The present invention also permits the creation of a set procedures for  
24 the sequence of steps to be followed. Thus, for example, a given advertisement can be  
25 appropriately constrained so that publication and/or editing of the advertisement

1 cannot occur until a credit check is obtained. Advantageously, and with particular  
2 applicability to advertising, the workflow management system preferably provides  
3 billing and publication tracking for a given advertisements to permit users to obtain  
4 up-to-the-minute information regarding the status of an advertisement. The workflow  
5 management system of the present invention may be appropriately modified to permit  
6 customized procedures and capabilities, on an individualized basis, for both  
7 advertising and editorial applications.

8       Traditionally, newsroom systems defined the progress of a story toward  
9 publication on the basis of the queue in which the story resided. To accomplish this  
10 functionality, the present invention provides the ability to define a system of  
11 automated statuses. These statuses are evaluated automatically when header fields  
12 change and are dynamically updated on all other user's systems that are currently  
13 viewing the same story or components. The evaluation of these statuses is also done  
14 using rule-based criteria, similar to the mechanism described previously for workflow  
15 steps.

16       In addition to automated queue movement and status changes, the workflow  
17 also provides a mechanism to execute scripted actions based on an item being placed  
18 into a "worklisted" queue. These queues can be set up to perform extensive and  
19 extensible actions using site configurable Visual Basic for Applications scripting  
20 language.

21       Alternatively, or in addition to, queues and/or status, the present invention  
22 provides "packages" to allows users to efficiently create stories in the database, assign  
23 their completion to other staff members and track their progress toward completion in  
24 terms of their relationship to users, not queues. Like other stories, packages have a  
25 header, but they differ in that they are assigned to users, not queues (discussed above).

1 Packages provide a convenient mechanism for the creation of story and photo  
2 assignments, the tracking of layouts and paginated pages pertaining to a collection of  
3 stories and the "clustering" of stories destined for non-print publication. Appendix B,  
4 attached herewith at the end of the Detailed Description, provides some exemplary  
5 header information for packages, which include data fields to achieve the stated  
6 functionality. Preferably, the system provides an appropriate user interface (e.g.,  
7 graphical user interface) to create and/or edit packages. By way of example,  
8 packages may be created and used as follows:

9       Editor Larry has been assigned the task of organizing his newspaper's  
10 coverage of the upcoming theater season. To simplify his work, Larry creates a  
11 package, which will  
12 include all elements destined to be published as well as supporting material he might  
13 need from his sub-editors and reporters. Via mechanisms provided in the package  
14 (Exhibit B) user interface, he creates the individual stories to be included in the  
15 package, automatically placing them in the appropriate reporter queues. He also  
16 generates photo assignments as part of creating the package, doling out work to the  
17 picture desk. During the weeks (if not months) it takes to prepare the package, he adds  
18 layout shapes he thinks might be used in the package. As production nears, he also  
19 adds the pages where the theater stories have been placed. During the daily news  
20 "huddles," Larry's editor is able to provide accurate information about the progress of  
21 the theater package by simply building a list of the packages assigned to Larry. With  
22 his previous editorial system, Larry would have had to hunt around myriad queues to  
23 check the status of the theater season coverage. With the present invention, all that is  
24 required is to query the system for the status of all the packages for which he is

1 responsible; the system immediately builds a description of the production status of all  
2 the package's components (and, necessarily, of the package itself).

3 Before the present invention is installed, city editor Max organized his  
4 reporters' workload by culling through piles of paper on his desk. Though he had a  
5 fairly good  
6 idea what his staff was working on, he preferably not could be sure of exactly what his  
7 people were doing. Making matters more difficult, the news editor recently began  
8 asking Max for productivity reports on the city desk staff. During the daily news  
9 huddle, Max is able to utilize the present invention to immediately describe the  
10 workload his staff will contribute to the next day's paper. It's no longer necessary to  
11 print and distribute paper news schedule; the present invention provides a convenient  
12 mechanism for querying the database about what's happening on the city desk. Max  
13 "horse trades" stories with other editors during the news huddle, immediately shifting  
14 the assignment of certain files from one package (and consequently one responsible  
15 editor) to another.

16 Based on what he learned in that day's news huddle, wire editor Rick knows  
17 what packages are being worked for the next day's paper. After the meeting, he  
18 returns to his desk and issues a query of all packages in process. Rick can see that  
19 Larry is working on a package regarding the upcoming theater season. He "offers" a  
20 story about a new play to  
21 Larry by automatically inserting it in Larry's package. Publications are free to  
22 generate editorial material without assigning it to a package and may conversely create  
23 packages that contain no links to material destined for publication.

24 Modifications of the present invention are also possible. For example, the  
25 unified data structure as described herein applies equally to the broadcast medium



1 (e.g. television and/or radio broadcast). Thus, for example, components can include  
2 video clips, audio/video presentations, audio sound bites, teleprompter text,  
3 commercial advertisements, or other data that is computer readable/transmittable or is  
4 otherwise computer controlled (e.g., analog video/audio equipment modified for  
5 computer control). In that regard, different insertions, as described above, can be  
6 applied for different geographic locations and/or different markets, while maintaining  
7 the single, unified data structure as described herein.

8 Of course, the system of the present invention can be adapted for other media  
9 outlets as well. For example, the system can be adapted as an editorial/advertisement  
10 story data management system for use by catalogs, magazines, newsletters, books,  
11 professional publications, inserts, directories, and/or other print media. As described  
12 above, the data management system of the present invention can track publishable  
13 stories (e.g., editorial stories and advertisement stories) across media channels, while  
14 maintaining the single, unified data structure as described herein. The data  
15 management system of the present invention can further be appropriately modified for  
16 use by news, advertising and similar agencies, independent of any publication  
17 operated by such agencies.

18 Further modifications are also possible. For example, the present invention  
19 can be appropriately modified to permit users to view/edit the publishable stories in  
20 multiple languages. To that end, the present invention can be adapted with language  
21 translation modules that permit any publishable story (and, accordingly, any insertions  
22 and components thereof) to be translated into a particular user's native language,  
23 edited, and then retranslated into the language as stored on the central database. In  
24 addition, the system of the present invention can be appropriately modified to  
25 incorporate exchange rates for multiple currencies, thereby permitting, for example,

1 users in different countries to run advertisements and charge the appropriate fee in the  
2 appropriate currency.

3 Advantageously, the data management system of the present invention  
4 maintains a rigorous enforcement of the relationship between publishable stories,  
5 insertions and components. By permitting different viewpoints of the data  
6 relationships, a user can perform collective operations on logical subsets of  
7 components or insertions, while the logical relationships between publishable story,  
8 insertion and component is maintained. Thus, rather than creating a new data file for  
9 each new instance of a publishable story, the present invention permits insertions and  
10 components to be reused unmodified in a different publication/medium/zone/edition,  
11 or modified as desired to accommodate the particulars of yet a different  
12 publication/medium/zone/edition, while maintaining the structural relationship  
13 between this data in a single, unified data structure. Thus, there is no need to create a  
14 new file for the same publishable story that is to appear in a different  
15 publication/medium/zone/edition. Advantageously, the system of the present  
16 invention applies the same set of global rules for both editorial and advertisement, and  
17 other stories. Additionally, the present invention provides a system that permits a user  
18 to make changes in an insertion and the system can automatically update the content  
19 of other insertions, thereby eliminating the need to make modifications in each  
20 insertion, and/or linking content so that a change in one publishable instance of that  
21 content will appear in all other linked instances of that content.

APPENDIX A**Component Attribute Information:**

In the present invention, components have an associated header, which essentially is a collection of data fields in which attributes can be user-defined or generated automatically. Some examples of preferred header data include:

**Author.** This field displays the system login name of the user responsible for the creation of the first version of the component's content. Author differs from creator in that a story may be created by a user who will have nothing to do with the content, as in the case of pagination user creating components as part of the page layout process or of a desk editor creating story assignments prior to the creation of page layouts. The Author field is preferably protected; it is preferably filled in by the system and cannot be modified by users.

**Character Count.** This field displays the number of characters contained in the component's text. This value is the sum of the non-white space characters in the component. The intent of this field is to provide a rough size estimate of the component but should not be considered a definitive measure of story size. The Character Count field is preferably conditionally locked; it cannot be updated while text with which it is associated is open for editing. In any event, the character count field is preferably protected from user modification. For non-textual components, this field will be left blank.

**Color.** This field is used to indicate whether a component has full or spot color. The user is free to enter the name of a spot color to be assigned to the component, though the value of the field will often be filled in by a pagination editor. The user will select either of the two options by selecting the appropriate option from a combo box. The

1 Color field is preferably not protected; it may be modified by the system or a user  
2 while another user has the content open for edit.

3 **Component Format.** This field is used to represent the format of the associated  
4 component. Possible entries in this field include (but is not limited to): audio  
5 components ( WAV, etc.), textual components (ASCII, XML, SGML, RTF, HTML,  
6 etc.), graphic components, including Images, ( JPG, GIF, EPS, TIFF, etc.), video  
7 components ( AVI, MOV). The contents of the Component Format field are  
8 preferably protected; the value is entered by the system when the component is  
9 created.

10 **Component Type.** This field is used to represent the type of information the  
11 component includes. Possible entries in this field include: audio, recurrent content,  
12 HTML, images,  
13 text, video, etc. The contents of the Component Type field are preferably protected;  
14 the value is entered by the system when the component is created.

15 **Copy Count.** This field displays the number of times this component has been copied.  
16 It is acceptable for this field to read either "0" or be blank in the headers of  
17 components which have not been copied. The Copy Count field is protected; it is  
18 preferably filled in by the system and cannot be modified by users.

19 **Cost/Payment.** This currency-formatted field is used to maintain dollar amounts for  
20 payment of free-lancers for a submission or the amount the newspaper will charge to  
21 publish the associated content. By default, this field will be left blank.  
22 The Cost/Payment field is preferably not protected; it may be modified by the system  
23 or a user while another user has the content open for edit.

24 **Creator.** This is the name of the user who created a component. Creator differs from  
25 Author in that the author is the first user to be responsible for a story's content — the

1 first user to modify the text, for example. The creator will often be a pagination user  
2 designing a page or an editor assigning stories to his or her reporters.  
3 The Creator field is protected; it is preferably filled in by the system and cannot be  
4 modified by users.

5 **Deadline Date and Time.** This unprotected date and time field displays when the  
6 component should have reached a specific point in the editing or pagination process.  
7 To streamline the entry of information into this field, a calendar combo box will be  
8 associated with this field as well as a streamlined mechanism for the entry of the time.  
9 By default, the Deadline Date and Time field will be blank. The Deadline Date and  
10 Time field is preferably not protected; it may be modified by the system or an  
11 authorized user while another user has the content open for edit.

12 **Editor.** This protected field displays the login name of the last user to edit the  
13 component's content. In instances where a user creates, edits then stores a newly-  
14 created component, the same login name will appear in the Author, Creator and Editor  
15 fields. The Editor field is protected; it is preferably filled in by the system and  
16 preferably cannot be modified by users.

17 **Editorial Status.** This field is used to specify the production status of a component  
18 from an editorial perspective. The Editorial Status field is preferably not protected; it  
19 may be modified by the system or an authorized user while another user has the  
20 content open for edit. The Editorial Status field entries of each component in an  
21 insertion are used to derive the editorial production status of the insertion to which the  
22 components belong. Essentially, the adage that a chain is as strong as its weakest link  
23 applies to editorial status information; the status assigned to the insertion is equivalent  
24 to the lowest status assigned to one of its components.

1 **Feature Modifier.** This field is used to specify information generated by the Con-  
2 figuration Manager to indicate that the story must be position relative to another  
3 editorial or advertising story. The Feature Modifier field is preferably not protected; it  
4 may be modified by the system or an authorized user while another user has the  
5 content open for edit. Entries in the Feature Modifier field can be generated by via  
6 conventional and/or proprietary display advertisement placement programs.

7 **Internal Reference Number.** Every story in the database will receive an internal  
8 reference number, distinguishing it from other stories. Though multiple versions of an  
9 element may share the same internal reference number (allowing the system to  
10 identify two modifications of a story as being related even though the other header  
11 information and text may differ considerably), no two different stories in the database  
12 may share the same internal reference number. The Internal Reference Number will be  
13 inserted into a field. The Internal Reference Number is protected; it is preferably  
14 filled in by the system and cannot be modified by users.

15 **Internet Status.** This field is used to specify the production status of a component  
16 from an Internet publishing perspective. The Internet Status field is preferably not  
17 protected; it may be modified by the system or an authorized user while another user  
18 has the content open for edit. The Internet Status field entries of each component in an  
19 insertion are used to derive the Web production status of the insertion to which the  
20 components belong. Essentially, the adage that a chain is as strong as its weakest link  
21 applies to Internet Status information; the status assigned to the insertion is equivalent  
22 to the lowest status assigned to one of its components.

23 **Jump From.** This field contains the identification of the page that the component  
24 jumps from. This identification should be translated to physical or logical page  
25 depending on whether the user has configured the system to show insertion

1 information using either physical or logical pages. This field is protected once a  
2 component is placed on a page and preferably can only be changed within a pagination  
3 editor.

4 **Jump Head.** This field contains the identification of the page on which a  
5 component's jump sequence begins. This identification should be translated to  
6 physical or logical page depending on how a user is referencing the component. This  
7 field is preferably protected. It is derived as the jump sequence is assigned.

8 **Jump Position.** This field contains the component's ordinal position in a jump  
9 sequence. This field is preferably protected. It is derived as the jump sequence is  
10 assigned.

11 **Jump To.** This field contains the identification of the page that the component jumps  
12 to. This identification should be translated to physical or logical page depending on  
13 how a user is referencing the component. This field is protected once a component is  
14 placed on a page and preferably can only be changed within the pagination software.

15 **Keywords.** This field displays keywords assigned to the insertion. The Keywords  
16 field is preferably not protected; it may be modified by the system or an authorized  
17 user while another user has the content open for edit.

18 **Locked By.** The name of the system user who currently has the content of the  
19 component locked. When a user locks the insertion, its associated components are also  
20 locked; the name of the user who locked the insertion is indicated in the component's  
21 Locked By field. Similarly, when a user locks the story, all components in all  
22 insertions are also locked. The Locked By field of all components and all insertions  
23 will be updated to reflect the login name of the user who has the story locked. In the  
24 event a user attempts to open for edit any story opened by another user (or the same  
25 user at another workstation), an error message will appear indicating the login name of

1 the user who has the story locked, the name of the workstation where the lock exists.

2 The user will be given the option of opening the story for read-only or aborting the

3 operation. If the user elects to open the story for read-only, it will automatically

4 become editable when the user editing the story releases the lock. The Locked By

5 field is protected; it is preferably filled in by the system and cannot be modified by

6 users.

7 **Locked By Workstation ID.** This field will be used for the storage of the workstation

8 ID where the component is locked. If the component is not locked, the field will be

9 blank. The Locked By Workstation ID field is protected; it is preferably filled in by

10 the system and cannot be modified by users.

11 **Makeup Date.** This unprotected date field displays the date in which the insertion

12 should go into the production process. To streamline the entry of information, a

13 calendar combo box will be associated with this field. The date is displayed using the

14 date configuration specified on the workstation. The Makeup Date field is preferably

15 not protected; it may be modified by the system or an authorized user while another

16 user has the content open for edit.

17 **Next Queue.** This unprotected field displays the name of the default next queue for

18 the component. The name of the Next Queue is entered in the administrative utilities

19 for the current queue and is displayed not only in the header for the component, but

20 also when the user invokes the Move command. If an authorized user modifies the

21 Next queue field entry, the revised queue name will appear when the user invokes the

22 Move command. So, for example, if the default move queue of a component residing

23 in queue A\B\C is X\Y\Z, but an editor changes the Next Queue field to be R\S\T,

24 R\S\T will appear as the default destination if the story is subsequently moved. The

25 Next Queue entry may also be overridden as part of the Move operation. The Next



1 Queue field is preferably not protected; it may be modified by the system or an  
2 authorized user while another user has the content open for edit. It is also possible for  
3 users to execute the move command to the Next Queue destination while another user  
4 has the content open for edit. A combo box with an editable entry field will be used  
5 for the selection of the next queue; all queues to which the user has write rights will  
6 be displayed when the combo box is activated. Assuming the user has the right to edit  
7 header information in the queue where the component resides, he or she may enter the  
8 name of a queue in the combo box entry field or select a queue from the list.

9 **Original Queue.** This field is used to indicate the queue where the component was  
10 created. The Original Queue field is protected; it is preferably filled in by the system  
11 and is preferably not modified by users.

12 **Pagination Status.** This field is used to specify the production status of a component  
13 from a pagination perspective. The Pagination Status field is preferably not protected;  
14 it may be modified by the system or an authorized user while another user has the  
15 content open for edit. The Pagination Status field entries of each component in an  
16 insertion are used to derive the Pagination Status of the insertion to which the  
17 components belong. Essentially, the adage that a chain is as strong as its weakest link  
18 applies to Pagination Status information; the status assigned to the insertion is  
19 equivalent to the lowest status assigned to one of its components.

20 **Parent Internal ID.** A unique ID of the parent story to allow a user to trace back from  
21 the child. In the case of components, the parent internal ID would refer to the internal  
22 ID of the component used as the source of the current component. The Parent Internal  
23 ID field is protected; it is preferably filled in by the system and is preferably not  
24 modified by users. No validation of the Parent Internal ID field will occur. Thus, if a

1 component's parent has been removed from the database, the Parent Internal ID field  
2 may include the ID of a story that no longer exists in the system database.

3 **Previous Editorial Status.** This field is used to specify the previous editorial  
4 production status of a component from an editorial perspective. The Previous Editorial  
5 Status field is protected; it is preferably filled in by the system and is preferably not  
6 modified by users.

7 **Previous Internet Status.** This field is used to specify the previous Internet Status of  
8 a component from an editorial perspective. The Previous Internet Status field is  
9 protected; it is preferably filled in by the system and is preferably not modified by  
10 users.

11 **Previous Pagination Status.** This field is used to specify the previous Pagination  
12 Status of a component from an editorial perspective. The Production Status field is  
13 protected; it is preferably filled in by the system and is preferably not modified by  
14 users.

15 **Previous Queue.** This protected field displays the name of the previous queue where  
16 the component resided. This field is initially blank until the component is moved from  
17 one queue to another. The Previous Queue field is protected; it is preferably filled in  
18 by the system and is preferably not modified by users.

19 **Priority.** This field displays information obtained from the Priority field in the  
20 standard NAA wire service (AP/UPI) header for the component. The priority field is  
21 unique in that sorts of the field are not necessarily alphabetical. In the United States  
22 for example, the priority field would be sorted in the following order: F (flash), B  
23 (bulletin), U (urgent), R (regular), D (deferred), H (hold), S (Sunday). The Priority  
24 field is preferably not protected; it may be modified by the system or an authorized  
25 user while another user has the content open for edit. The Priority field will be

1 displayed to the user as a combo box with an editable entry field. The user may either  
2 type the letter corresponding to the desired priority or select an entry from the  
3 resulting list.

4 **Queue.** This field displays the queue in which the component currently resides.  
5 Components are the only type of stories which, from the user's perspective, appear to  
6 reside in queues. The Queue field may be modified by the user who has the content of  
7 a component locked, or via the user interface by an authorized user without first  
8 locking the content. It will not be possible for one user to modify the Queue field of a  
9 component if the content is locked by another user (or by the same user at a different  
10 workstation). The Queue field will be represented to the user as a combo box with an  
11 editable entry field. When selected, the box will display all the queues to which the  
12 user has at least the right to write content. Assuming the user has the right to edit  
13 header information, he or she will be free to enter any of the values "by hand" or  
14 select one of the user names from the combo box. This field would preferably only  
15 apply to editorial events, not advertising events, since advertising events typically are  
16 not associated with queues.

17 **Responsibility.** This field displays the login name of the system user responsible for  
18 ensuring that the production of this component takes place successfully. The  
19 Responsibility field is preferably not protected; it may be modified by the system or an  
20 authorized user while another user has the content open for edit. The Responsibility  
21 field will be represented to the user as a combo box with an editable entry field. When  
22 selected, the box will display all system users. The user will be free to enter any of the  
23 values "by hand" or select one of the user names from the combo box.

24 **Responsible Department or Desk.** This field displays the name of the department or  
25 desk responsible for ensuring that the production of this component takes place

1 successfully. The Responsible Department or Desk field is preferably not protected; it  
2 may be modified by the system or an authorized user while another user has the  
3 content open for edit. The Responsible Department or Desk field will be represented  
4 to the user as a combo box with an editable entry field. When selected, the box will  
5 display all of the publication's departments. The user will be free to enter any of the  
6 values "by hand" or select one of the departments from the combo box.

7 **Slug.** This field contains the name of the component. Component names roughly  
8 correspond to the name of NAA SGML tags for incoming wire material. The Slug  
9 field for components will be protected; possible names are configured upon system  
10 installation. To create a component slug, the user will select the type of component  
11 when creating the component. The system will automatically number the story type  
12 depending on how many similar stories are in the story; component names are unique  
13 across all insertions for a particular story.

14 **Time Arrived in Queue.** This date field displays the time when a particular version  
15 of the component arrived in its current queue. The date and time are displayed using  
16 the date configuration specified on the workstation. The Time Arrived in Queue field  
17 is protected; it is preferably filled in by the system and is preferably not modified by  
18 users. This field would only apply to editorial events, not advertising events, since  
19 advertising events typically are not associated with queues.

20 **Time Created.** This date field displays the time when a particular component was  
21 created. The date and time are displayed using the date configuration specified on the  
22 workstation. The Time Created field is protected; it is preferably filled in by the  
23 system and preferably not modified by users.

24 **Time Modified.** This date field displays the time the component was created. The  
25 date and time are displayed using the date configuration specified on the workstation.

1 The Time Modified field is protected; it is preferably filled in by the system and  
2 preferably not modified by users.

3 **Version.** This field displays the version number of the component. The Version field  
4 is protected; it is preferably filled in by the system and preferably not modified by  
5 users.

6 **Insertion Attribute Information:**

7 In the present invention, insertions have an associated header, which  
8 essentially is a collection of data fields in which attributes can be user-defined or  
9 generated automatically. Some examples of preferred header data include:

10 **Absolute Page.** This field displays the absolute page to which the insertion has been  
11 assigned. The pagination software is capable of updating this field as stories are  
12 assigned to pages. In addition to the absolute page values specified in the  
13 Configuration Manager, the Absolute Page field may be set to TBD — “To Be  
14 Determined.” The Absolute Page field will preferably be protected; Absolute Page  
15 information for a component may be changed either by altering the absolute page field  
16 for the insertion to which it belongs, by “dragging and dropping” the component into  
17 another insertion or by manual (user-supplied) commands.

18 **Author.** This field displays the login name of the user first responsible for the creation  
19 of an insertion. Author differs from creator in that a story may be created by a user  
20 who will have nothing to do with the content, as in the case of a pagination user  
21 creating components as part of the page layout process or of a desk editor creating  
22 story assignments prior to the creation of page layouts. The name of the user who first  
23 modifies the content is the author. The Author field is protected; it is preferably filled  
24 in by the system and cannot be modified by users.

1    **Block Count.** This field displays the contents of the NAA Block Count field as  
2    received from incoming wire services for a particular story. The block count field is  
3    not used on stories generated "in-house" (i.e., locally, as opposed to generated via  
4    wire service) — it will be empty in the headers of stories generated by users. The  
5    block count field is entered into the insertion header created as part of the reception of  
6    a story from a wire service; it is left blank in the insertion headers of stories created by  
7    copying the initial insertion. The Block Count field is protected; it is preferably filled  
8    in by the system and is preferably not modified by users.

9    **Category.** This field displays information obtained from the Category field in the  
10   standard NAA wire service header for the story. Generally, the category field is left  
11   blank on stories generated in-house, though customers are free to use it. The category  
12   field is inserted into the headers of stories transmitted from the newspaper.

13   The Category field is preferably not protected; it may be modified by the system or an  
14   authorized user while another user has the content open for edit.

15   **Copy Count.** This protected field displays the number of times this insertion has been  
16   copied. It is acceptable for this field to read either "0" or be blank in the headers of  
17   insertions which have not been copied. The Copy Count field is protected; it is  
18   preferably filled in by the system and cannot be modified by users.

19   **Cost/Payment.**<sup>1</sup> This currency-formatted field is used to maintain dollar amounts for  
20   payment of free-lancers for a submission or the amount the newspaper will charge to  
21   publish the associated content. The information included in this field is the sum of the  
22   Cost/Payment fields of each of the components included in the insertion. The  
23   Cost/Payment field is preferably not protected; it may be modified by the system or a  
24   user while another user has the content open for edit.

1   **Creator.** This is name of the user who created an insertion. Creator differs from  
2   Author in that the author is the first user to be responsible for a story's content — the  
3   first user to modify the text, for example. The creator will often be a pagination user  
4   designing a page or an editor assigning stories to his or her reporters. The Creator  
5   field is protected; it is preferably filled in by the system and cannot be modified by  
6   users.

7   **Deadline Date and Time.** This unprotected date and time field displays when the  
8   insertion should have reached a specific point in the editing or pagination process. To  
9   streamline the entry of information into this field, a calendar combo box will be used  
10   as well as a streamlined mechanism for the entry of time. By default, the Deadline  
11   Date and Time field will be blank. The Deadline Date and Time field is protected; it  
12   is preferably filled in by the system and cannot be modified by users. Though the  
13   components that make up an insertion may have information in their Deadline Date  
14   and Time field, there is no correlation between component deadlines and the deadline  
15   of the insertion.

16   **Edition.** This protected field displays the edition in which the insertion is intended to  
17   be published. Default publication sizing information, gleaned from data maintained  
18   by the Configuration Manager, will be displayed. Modifications to the Edition field of  
19   the insertion necessarily modifies the Edition field of all components it contains. The  
20   Edition field will be represented in the insertion header as a combo box with an  
21   associated editable entry field. The user will either type the name of a valid edition or  
22   may select one from the combo box. The Edition field is linked to the Publication,  
23   Publication Date, Zone, Logical Page, Logical Section, Physical Page, Physical  
24   Section and Absolute Page fields in that a modification to the Edition field may result  
25   in the entries in the other fields to change to reflect valid publication information. In

1 addition to the editions specified in the Configuration Manager, the Edition field may  
2 be set to TBD — “To Be Determined.” The Edition field may be modified via the  
3 insertion header, by altering a displayed field in the Directory Manager or by invoking  
4 manual commands.

5 **Editorial Status.** This field is used to indicate the “lowest” Editorial Status of a  
6 component belonging to the insertion. Essentially, the adage that a chain is as strong  
7 as its weakest link applies to the Editorial Status information; the status assigned to  
8 the insertion is equivalent to the lowest status assigned to one of its components. The  
9 Editorial Status field is preferably protected; its contents are either derived from the  
10 production statuses of the components contained in the insertion or is set by  
11 pagination software.

12 **Feature Modifier.** This field is used to specify information generated by the  
13 Configuration Manager to indicate that the story must be position relative to another  
14 editorial or advertising story. The Feature Modifier field is preferably not protected; it  
15 maybe modified by the system or an authorized user while another user has the  
16 content open for edit. It is anticipated that entries in the Feature Modifier field will be  
17 generated by display advertisement placement programs via the Page One third- party  
18 interface API.

19 **Internal Reference Number.** Every story in the database will receive an internal  
20 reference number, distinguishing it from other stories. Though multiple versions of an  
21 element may share the same internal reference number (allowing the system to  
22 identify two modifications of a story as being Document Name: related even though  
23 the other header information and text may differ considerably), no two different  
24 stories in the database may share the same internal reference number. The Internal



1 Reference Number will be inserted into a field. The Internal Reference Number is  
2 protected; it is preferably filled in by the system and cannot be modified by users.

3 **Internet Status.** This field is used to indicate the “lowest” Internet Status of a  
4 component belonging to the insertion. Essentially, the adage that a chain is as strong  
5 as its weakest link applies to the Internet Status information; the status assigned to the  
6 insertion is equivalent to the lowest status assigned to one of its components. The  
7 Internet Status field for insertions is preferably protected; its contents are either  
8 derived from the production statuses of the components contained in the insertion or is  
9 set by pagination software.

10 **Locked By.** The name of the user who currently has the insertion locked —the user  
11 who has all components locked. In the event a user attempts to open for edit any story  
12 opened by another user (or the same user at another workstation), an error message  
13 will appear indicating the login name of the user who has the story locked and the  
14 name of the workstation where the lock exists. The user will be given the option of  
15 opening the story for read-only, “taking a number” for a cascading lock or aborting the  
16 operation. If the user elects to “take a number,” it will automatically become editable  
17 when the user editing the story releases the lock. The Locked By field is protected; it  
18 is preferably filled in by the system and cannot be modified by users.

19 **Locked By Pagination Software.** This field will contain the name of the user who  
20 has locked the last unlocked component. Once a story has been placed on a page, its  
21 position and geometry may not be modified outside of the pagination software until  
22 the lock is released. This field is protected and may only be modified by a pagination  
23 session or an authorized user.

24 **Locked By Workstation ID.** This field will be used for the storage of the workstation  
25 ID of the user who has all components of the insertion locked. If any of the

1 components are locked by any other user, this field will be set to "Locked by Multiple  
2 Users." If no components are locked, the field will be blank. The Locked By  
3 Workstation ID field is protected; it is preferably filled in by the system and cannot be  
4 modified by users.

5 **Logical Page.** This field displays the logical page to which the insertion has been  
6 assigned. Default publication sizing information, gleaned from data maintained by the  
7 Configuration Manager, will be displayed. Modifications to the logical pages field of  
8 the insertion necessarily modify the Logical Page fields of all components it contains.  
9 The Logical Page field will be represented in the insertion header as a combo box with  
10 an associated editable entry field. The user will either type the name of a valid logical  
11 page or may select one from the combo box. The Logical Page field is linked to the  
12 Publication, Publication Date, Zone, Edition, Logical Section, Physical Page, Physical  
13 Section and Absolute Page fields in that a modification to the Logical Page field may  
14 result in the entries in the other fields to change to reflect valid publication  
15 information. In addition to the logical pages specified in the Configuration Manager,  
16 the Logical Page field may be set to TBD — "To Be Determined." The Logical Page  
17 field may be modified via the insertion header, by altering a displayed field in the  
18 Directory Manager or by invoking manual commands.

19 **Makeup Date.** This unprotected date field displays the date in which the insertion  
20 should go into the production process. To streamline the entry of information into this  
21 field, a calendar combo box will be associated with this field. The date is displayed  
22 using the date configuration specified on the workstation. The system must be capable  
23 of evaluating the Makeup Date of the insertion in relation to the makeup date of its  
24 associated components. If the user enters a makeup date in the insertion that is before  
25 the makeup date for any of its corresponding components, an information dialog box

1 will appear with the message "The makeup date for the insertion will occur before the  
2 makeup date for the component slug component." This dialog box will appear for  
3 each component whose makeup date is later than the makeup date of the insertion.  
4 The Makeup Date field is preferably not protected; it may be modified by the system  
5 or an authorized user while another user has the content open for edit.

6 **Original Slug.** This field displays the slug originally assigned to the insertion. The  
7 name of the insertion will be derived from its run information; users will not  
8 "manually" assign slugs to insertions. The Original Slug field is protected; it is  
9 preferably filled in by the system and is preferably not modified by users.

10 **Output Medium Type.** The output medium type for an insertion indicates the type of  
11 medium where the components of the insertion will be published. It is not possible to  
12 assign invalid components to an insertion. This is, on the basis of the output medium  
13 type, only a particular set of components may be used. The Output Medium Type field  
14 is protected; it is preferably filled in by the system and is preferably not modified by  
15 users.

16 **Pagination Status.** This field is used to indicate the "lowest" Pagination Status of a  
17 component belonging to the insertion. The Pagination Status field is preferably not  
18 protected; it may be modified by the system or an authorized user while another user  
19 has the content open for edit.

20 **Physical Page.** This field displays the physical page to which the insertion has been  
21 assigned. Default publication sizing information, gleaned from data maintained by the  
22 Configuration Manager, will be displayed. Modifications to the physical pages field of  
23 the insertion necessarily modify the Physical Page fields of all components it contains.  
24 As with all other modifications to the run information header fields, modifications to  
25 one field may cause the information to be displayed in another field to automatically

1 change. The Physical Page field is linked to the Publication, Publication Date, Zone,  
2 Edition, Logical Section, Logical Page, Physical Section and Absolute Page fields in  
3 that a modification to the Physical Page field may result in the entries in the other  
4 fields to change to reflect valid publication information. In addition to the physical  
5 pages specified in the Configuration Manager, the Physical Page field may be set to  
6 TBD — “To Be Determined.”

7 The Physical Page field will preferably be protected; Physical Page information for an  
8 insertion may be changed by invoking the Assign and/or Place commands.

9 **Placement Priority.** This field will be used as an indication of the importance for an  
10 insertion to appear in a particular position. It is anticipated that it will be used  
11 primarily by the algorithmic editorial pagination placement utility. The Placement  
12 Priority field is preferably not protected; it may be modified by the system or an  
13 authorized user while another user has the content open for edit.

14 **Previous Editorial Status.** This field is used to indicate the previous Editorial Status  
15 of the insertion. The Previous Editorial Status field is protected; it is preferably filled  
16 in by the system and cannot be modified by users.

17 **Previous Internet Status.** This field is used to indicate the previous Internet Status of  
18 the insertion. The Previous Internet Status field is protected; it is preferably filled in  
19 by the system and cannot be modified by users.

20 **Previous Pagination Status.** This field is used to indicate the previous Pagination  
21 Status of the insertion. For a description of all possible entries, see the Pagination  
22 Status section of chapter 6. The Previous Pagination Status field is protected; it is  
23 preferably filled in by the system and cannot be modified by users.

24 **Priority.** This field displays information obtained from the Priority field in the  
25 standard NAA wire service header for the insertion. The priority field is unique in that

1 sorts of the field are not necessarily alphabetical. In the United States for example, the  
2 priority field would be sorted in the following order: F (flash), B (bulletin), U  
3 (urgent), R (regular), D (deferred), H (hold), S (Sunday). The sort order of the priority  
4 field is validated against a configuration file maintained in the database. The Priority  
5 field is preferably not protected; it may be modified by the system or an authorized  
6 user while another user has the content open for edit.

7 **Publication.** This field indicates the target publication for the insertion. Information  
8 in this field may be derived by the configuration information maintained in  
9 Configuration Manager. Modifications to the Publication field of the insertion  
10 necessarily modify the Publication field of all components it contains.

11 The Publication field is linked to the Publication Date, Zone, Edition, Logical Page,  
12 Logical Section, Physical Page, Physical Section and Absolute Page fields in that a  
13 modification to the Publication field may result in the entries in the other fields to  
14 change to reflect valid publication information. In addition to the publication specified  
15 in the Configuration Manager, the Publication field may be set to TBD — “To Be  
16 Determined.” The Publication field will preferably be protected; publication  
17 information for an insertion may be changed by invoking manual commands.

18 **Publication Date.** This field displays the date when the insertion is expected to be  
19 published. The date is displayed using the date configuration specified on the  
20 workstation. Modifications to the publication date field of the insertion necessarily  
21 modify the Publication Date field of all components it contains. The Publication Date  
22 field is linked to the Publication, Zone, Edition, Logical Page, Logical Section,  
23 Physical Page, Physical Section and Absolute Page fields in that a modification to the  
24 Publication

1 Date field may result in the entries in the other fields to change to reflect valid  
2 publication information. In addition to the publication dates specified in the  
3 Configuration Manager, the Publication Date field may be set to TBD - "To Be  
4 Determined." The Publication Date field will preferably be protected; publication date  
5 information for an insertion may be changed by invoking manual commands.

6 **Responsible Department or Desk.** This field displays the name of the department or  
7 desk responsible for ensuring that the production of this insertion takes place  
8 successfully. The Responsible Department or Desk field is preferably not protected; it  
9 may be modified by the system or an authorized user while another user has the  
10 insertion's components open for edit.

11 **Slug.** This field contains the name of the insertion. Insertion names are derived from  
12 configuration information established on system installation; some publications may  
13 choose to organize insertion slugs on the basis of logical page numbers, others may  
14 choose to do so on the basis of physical page configurations. The following example  
15 presumes the customer chose to organize insertion slugs on the basis of logical page  
16 information. Information is presented in the following format: logicalpage,  
17 logicalsection, pub, pubdate, edition, zone as in 3,Sports,Star,24-Mar-1999,5,N  
18 Moving the mouse pointer over the page information in an insertion slug will cause a  
19 "balloon help" window to appear showing the physical and absolute page equivalents  
20 of the logical page assignment. In systems configured to show page numbers in  
21 relation to the physical page, moving the mouse pointer over the page information in  
22 an insertion slug will cause a balloon help window to appear showing the logical and  
23 absolute page equivalents of the physical page assignment. The Slug field for  
24 insertions is protected; it is preferably filled in by the system and preferably not  
25 modified by users.

- 1   **Time Created.** This date field displays the time when a particular insertion was  
2   created. The date and time are displayed using the date configuration specified on the  
3   workstation.
- 4   The Time Created field is protected; it is preferably filled in by the system and  
5   preferably not modified by users.
- 6   **Time Modified.** This date field displays the time the last component of the insertion  
7   was modified. The date and time are displayed using the date configuration specified  
8   on the workstation. The Time Modified field is protected; it is preferably filled in by  
9   the system and preferably not modified by users.
- 10   **X Position** field represents the position on the X axis of the upper left-hand corner of  
11   the component. For the insertion, it is the upper left-hand corner of the overall  
12   bounding box. This field is protected if the component or insertion is locked by the  
13   pagination editor or has a status of Placed or higher. When the story is locked by the  
14   pagination editor, only the session holding the lock may change the position. When  
15   the status is greater than Placed, the position may only be changed by the pagination  
16   editor.
- 17   **Y Position** field represents the position on the Y axis of the upper left-hand corner of  
18   the component. For the insertion, it is the upper left-hand corner of the overall  
19   bounding box. This field is protected if the component or insertion is locked by the  
20   pagination editor or has a status of Placed or higher. When the story is locked by the  
21   pagination editor, only the session holding the lock may change the position. When  
22   the status is greater than Placed, the position may only be changed by the pagination  
23   editor.
- 24   **Zone.** This field displays the zone in which the insertion is intended to be published.

1 Modifications to the Zone field of the insertion necessarily modify the Zone fields of  
2 all components it contains. The Zone field is linked to the Publication, Publication  
3 Date, Physical Page, Physical Section, Edition, Logical Page, Logical Section and  
4 Absolute Page fields in that a modification to the Zone field may result in the entries  
5 in the other fields to change to reflect valid publication information. In addition to the  
6 zones specified in the Configuration Manager, the Zone field may be set to TBD —  
7 “To Be Determined.” The Zone field will preferably be protected; zone information  
8 for an insertion may be changed by invoking manual commands.

9 **Story Attribute Information:**

10 In the present invention, stories have an associated header, which essentially is  
11 a collection of data fields in which attributes can be user-defined or generated  
12 automatically. A story is a collection of insertions. For publication, a single story  
13 includes at least one insertion and there is no limit to the number of insertions that  
14 may be included in a story. Stories may enter the system by wire services or another  
15 third-party authoring mechanism, or, of course, by reporters and editors, or other  
16 sources. A story may also be generated by parsing an incoming graphic, splitting its  
17 caption information into a discrete database story apart from the image itself. Some  
18 examples of preferred header data include:

19 **Author.** This field displays the login name of the user first responsible for the creation  
20 of content for the story. Author differs from creator in that a story may be created by a  
21 user who will have nothing to do with the content, as in the case of a pagination user  
22 creating stories as part of the page layout process. The Author field is protected; it is  
23 preferably filled in by the system and cannot be modified by users.

24 **Copy Count.** This protected field displays the number of times this story has been  
25 copied. It is acceptable for this field to read either “0” or be blank in the headers of



1 components which have not been copied. The Copy Count field is protected; it is  
2 preferably filled in by the system and cannot be modified by users.

3 **Cost/Payment.** 1 This currency-formatted field is used to maintain dollar amounts for  
4 payment of free-lancers for a submission or the amount the newspaper will charge to  
5 publish the associated content. By default, the story Cost/Payment field is the sum of  
6 the Cost/Payment fields of each of the insertions contained in the story. However, the  
7 Cost/Payment field is preferably not protected; it may be modified by the system or a  
8 user while another user has the content open for edit.

9 **Creator.** This is the name of the user who created the story. Creator differs from  
10 Author in that the author is the first user to be responsible for a story's content. The  
11 creator will often be a pagination user designing a page. The Creator field is  
12 protected; it is preferably filled in by the system and cannot be modified by users.

13 **Cycle Designator.** This field is used to indicate the cycle a wire story was transmitted.  
14 This information is parsed from the NAA keyword field and are typically either AM-,  
15 PM- or BC-. This field is normally left blank on stories not generated by wire  
16 services. The Cycle Designator field is preferably not protected; it may be modified by  
17 an authorized user at any time.

18 **Deadline Date and Time.** Content for this protected date and time field will be  
19 derived from the story's insertion with the latest deadline. The Deadline Date and  
20 Time field is protected; it is preferably filled in by the system and cannot be modified  
21 by users

22 **Editor.** This protected field displays the login name of the last user to edit the content  
23 of any components belonging to an insertion assigned to the story. The Editor field is  
24 protected; it is preferably filled in by the system and cannot be modified by users.

- 1    **Editorial Status.** This field is used to indicate the “lowest” Editorial Status of an  
2    insertion belonging to the story. The Editorial Status field is protected; it is preferably  
3    filled in by the system and cannot be modified by users.
- 4    **Feature Modifier.** This field is used to specify information generated by the Con-  
5    figuration Manager to indicate that the story must be position relative to another  
6    editorial or advertising story. The Feature Modifier field is preferably not protected; it  
7    may be modified by the system or an authorized user while another user has the  
8    content open for edit.
- 9    **Internal Reference Number.** Every story in the database will receive an internal  
10    reference number, distinguishing it from other stories. Though multiple versions of a  
11    story may share the same internal reference number (allowing the system to identify  
12    two modifications of a story as being related even though the other header information  
13    and text may differ considerably), no two different stories in the database may share  
14    the same internal reference number. The Internal Reference Number will be inserted  
15    into a field. The Internal Reference Number is protected; it is preferably filled in by  
16    the system and cannot be modified by users.
- 17    **Internet Status.** This field is used to indicate the “lowest” Internet Status of an  
18    insertion belonging to the story. The Internet Status field is protected; it is preferably  
19    filled in by the system and cannot be modified by users.
- 20    **Keywords.** This unprotected field displays keywords assigned to the story. The  
21    Keywords field is preferably not protected; it may be modified by the system or an  
22    authorized user while another user has the content open for edit.
- 23    **Locked By.** The name of the user who currently has the story locked. When a user  
24    locks the story, all insertions and components it includes are also locked. Any  
25    components

1 already opened for editing when the user attempts to open the story will be opened in  
2 read-only mode and, if the user has edit rights in the queues where the locked  
3 components reside, he or she will be given the option of "taking a number," which  
4 indicates to the system that the lock should be cascaded when the story is unlocked by  
5 the previous user. The Locked By field is protected; it is preferably filled in by the  
6 system and cannot be modified by users.

7 **Locked By Workstation ID.** This field will be used for the storage of the workstation  
8 ID where the story is locked. If the story is not locked, the field will be blank. The  
9 Locked By Workstation ID field is protected; it is preferably filled in by the system  
10 and cannot be modified by users.

11 **Makeup Date.** This unprotected date field displays the date in which the story should  
12 go into the production process. To streamline the entry of information into this field, a  
13 calendar combo box will be associated with this field. The date is displayed using the  
14 date configuration specified on the workstation. The Makeup Date field is preferably  
15 not protected; it may be modified by the system or an authorized user while another  
16 user has the content open for edit.

17 **Original Slug.** This field displays the slug originally assigned to the story. Incoming  
18 wire stories will be named using the fields that correspond to the NAA and IPTC  
19 keyword entries. The Original Slug field is protected; it is preferably filled in by the  
20 system and is preferably not modified by users.

21 **Pagination Status.** This field is used to indicate the "lowest" Pagination Status of an  
22 insertion belonging to the story. The Pagination Status field is protected; it is  
23 preferably filled in by the system and cannot be modified by users. Optionally, the  
24 cycle designator will be stripped from the slug. Note that because wire services

1 routinely repeat key-words, components of stories with the same slug will exist in a  
2 single queue.

3 **Parent Internal ID.** A unique ID of the parent story to allow a user to trace back from  
4 the child. In the case of stories, the parent internal ID would refer to the internal ID of  
5 the story used as the source of the current story. The Parent Internal ID field is  
6 protected; it is preferably filled in by the system and is preferably not modified by  
7 users. Preferably, no validation of the Parent Internal ID field will occur. Thus, if a  
8 story's parent has been removed from the database, the Parent Internal ID field may  
9 include the ID of a story that no longer exists in the system database.

10 **Previous Editorial Status.** This field is used to indicate the previous Editorial Status  
11 of a component belonging to the insertion. The Previous Editorial Status field is  
12 protected; it is preferably filled in by the system and cannot be modified by users.

13 **Previous Internet Status.** This field is used to indicate the previous Internet Status of  
14 an insertion belonging to the story. The Previous Internet Status field is protected; it is  
15 preferably filled in by the system and cannot be modified by users.

16 **Previous Pagination Status.** This field is used to indicate the previous Pagination  
17 Status of an insertion belonging to the story. The Previous Pagination Status field is  
18 protected; it is preferably filled in by the system and cannot be modified by users.

19 **Priority.** This field displays information obtained from the Priority field in the  
20 standard NAA wire service header for the story. The priority field is unique in that  
21 sorts of the field are not necessarily alphabetical. In the United States for example, the  
22 priority field would be sorted in the following order: F (flash), B (bulletin), U  
23 (urgent), R (regular), D (deferred), H (hold), S (Sunday). The sort order of the priority  
24 field is validated against a configuration file maintained in the database. The Priority

1 field is preferably not protected; it may be modified by the system or an authorized  
2 user while  
3 another user has the content open for edit.

4 **Responsibility.** This field displays the login name of the user responsible for ensuring  
5 that the production of this story takes place successfully. The Responsibility field is  
6 preferably not protected; it may be modified by the system or an authorized user while  
7 another user has the content open for edit.

8 **Responsible Department or Desk.** This field displays the name of the department or  
9 desk responsible for ensuring that the production of this story takes place successfully.  
10 The Responsible Department or Desk field is preferably not protected; it may be  
11 modified by the system or an authorized user while another user has the content open  
12 for edit.

13 **Slug.** This field contains the slug of the story. For incoming wire material, the key-  
14 word field of the NAA 1312 specification or IPTC header equivalent will be used as  
15 the name; duplicates will be allowed. Story Slug fields are preferably unprotected.  
16 Valid characters for slugs include upper- and lower-case letters A to Z, numbers zero  
17 to nine, the period, comma, spaces, hyphens and underscores.

18 **Time Created.** This date field displays the time when a particular story was created.  
19 The date and time are displayed using the date configuration specified on the  
20 workstation.

21 The Time Created field is protected; it is preferably filled in by the system and  
22 preferably not modified by users.

23 **Time Modified.** This date field displays the time any component or insertion of a  
24 story was modified. The date and time are displayed using the date configuration

- 1 specified on the workstation. The Time Modified field is protected; it is preferably
- 2 filled in by the system and preferably not modified by users.
- 3 **Wire Reference Number.** This field displays the contents of the NAA Slug field.
- 4 The Wire Reference Number field is protected; it is preferably filled in by the system
- 5 and is preferably not modified by users.

**APPENDIX B**

In the present invention, packages have an associated header, which essentially is a collection of data fields in which attributes can be user-defined or generated automatically. A package defines a mechanism by which stories are defined and how stories are tracked throughout the entire process: from inception to final destination. Packages are the preferred mechanism for workflow management of the present invention. Some examples of preferred header data include:

**Author.** This field displays the login name of the user responsible for the creation of a package. Author differs from creator in that a story may be created by a user who will have nothing to do with the content, as in the case of a pagination user creating components as part of the page layout process or of a desk editor creating story assignments prior to the creation of page layouts. The name of the user who first modifies the content is the author. The Author field is protected; it is preferably filled in by the system and cannot be modified by users.

**Category.** Editors may enter category information pertaining to the package in this field. The Category field is preferably not protected; it may be modified by the system or an authorized user while another user has the content open for edit.

**Copy Count.** This protected field displays the number of times the package has been copied. It is acceptable for this field to read either "0" or to be blank in the headers of packages which have not yet been copied. The Copy Count field is protected; it is preferably filled in by the system and cannot be modified by users.

**Creator.** This is the name of the user who created a package. Creator differs from Author in that the author is the first user to be responsible for a story's content — the first user to modify a component's text, for example. The creator will often be a pagination user designing a page or an editor assigning stories to his or her reporters.

- 1 The Creator field is protected; it is preferably filled in by the system and cannot be  
2 modified by users.
- 3 **Editor.** This protected field displays the login name of the last user to edit the  
4 package. In instances where a user creates, edits, then stores a newly-created package,  
5 the same login name will appear in the Author, Creator and Editor package header  
6 fields. The Editor field is protected; it is preferably filled in by the system and cannot  
7 be modified by users.
- 8 **Feature Modifier.** This field is used to specify information generated by the  
9 Configuration Manager to indicate that the story must be position relative to another  
10 editorial or advertising story. The Feature Modifier field is preferably not protected; it  
11 may be modified by the system or an authorized user while another user has the  
12 content open for edit. It is anticipated that entries in the Feature Modifier field will be  
13 generated by display advertisement placement programs via, for example, an interface  
14 API.
- 15 **Internal Reference Number.** Every story in the database will receive an internal  
16 reference number, distinguishing it from other stories. Though multiple versions of an  
17 element may share the same internal reference number (allowing the system to  
18 identify two modifications of a story as being related even though the other header  
19 information and text may differ considerably), no two different stories in the database  
20 may share the same internal reference number. The Internal Reference Number will be  
21 inserted into a field. The Internal Reference Number is protected; it is preferably  
22 filled in by the system and cannot be modified by users.
- 23 **Keywords.** This unprotected field displays keywords assigned to the package. The  
24 Keywords field is preferably not protected; it may be modified by the system or an  
25 authorized user while another user has the content open for edit.



- 1   **Original Slug.** This field displays the slug originally assigned to the package. Like  
2   stories, packages are one of the few types of stories where the user generates the slug.  
3   The Original Name field is protected; it is preferably filled in by the system and is  
4   preferably not modified by users.
- 5   **Parent Internal ID.** A unique ID of the parent story to allow a user to trace back from  
6   the child. In the case of packages, the parent internal ID would refer to the internal ID  
7   of the package used as the source of the current package. The Parent Internal ID field  
8   is protected; it is preferably filled in by the system and is preferably not modified by  
9   users.
- 10   **Parent Slug.** This field displays the slug of the package from which the current pack-  
11   age was copied. The Parent Name field is protected; it is preferably filled in by the  
12   system and is preferably not modified by users.
- 13   **Priority.** This priority field is unique in that sorts of the field are not necessarily  
14   alphabetical. In the United States for example, the priority field would be sorted in the  
15   following order: F (flash), B (bulletin), U (urgent), R (regular), D (deferred), H (hold),  
16   S (Sunday). The sort order of the priority field is validated against a configuration file  
17   maintained in the database. The Priority field is preferably not protected; it may be  
18   modified by the system or an authorized user while another user has the content open  
19   for edit.
- 20   **Responsibility.** This field displays the login name of the user responsible for ensuring  
21   that the production of this package takes place successfully. The Responsibility field  
22   is preferably not protected; it may be modified by the system or an authorized user  
23   while another user has the content open for edit.
- 24   **Responsible Department or Desk.** This field displays the name of the department or  
25   desk responsible for ensuring that the production of this package takes place

- 1 successfully. The Responsible Department or Desk field is preferably not protected; it  
2 may be modified by the system or an authorized user while another user has the  
3 content open for edit.
- 4 **Slug.** This field contains the slug of the package. The Slug field is preferably  
5 unprotected with packages.
- 6 **Time Created.** This date field displays the time when a package was created. The  
7 date and time are displayed using the date configuration specified on the workstation.  
8 The Time Created field is protected; it is preferably filled in by the system and  
9 preferably not modified by users.
- 10 **Time Modified.** This date field displays the time the most-recently modified  
11 component was stored back to the database. The date and time are displayed using the  
12 date configuration specified on the workstation. The Time Modified field is protected;  
13 it is preferably filled in by the system and preferably not modified by users.

CLAIMS

- 1  
2 1. A data management and editorial system, comprising: a central database for  
3 storing at least one publishable story, said story comprising data content and at least  
4 one insertion related to said data content, said insertion including one or more  
5 attributes related to said data content wherein at least one attribute defines a  
6 destination for said content, said central database being adapted to relate the content  
7 and insertions of said story in a unified data structure to permit said insertions and/or  
8 content of said story to be viewed and/or edited.
- 9 2. A system as claimed in claim 1, wherein said publishable story includes a  
10 publishable event for a print medium, broadcast medium or Internet medium.
- 11 3. A system as claimed in claim 1, wherein said content includes text, graphics,  
12 video and/or audio data that is computer readable/transmittable.
- 13 4. A system as claimed in claim 1, further comprising an *a priori* database which  
14 is used to define preset data related to said story, insertion, and/or content.
- 15 5. A system as claimed in claim 1, wherein each instance of said content being  
16 given an associated content header, wherein said header includes data fields related to  
17 the format of said content, wherein at least one of said data fields being used to  
18 associate said content with at least one said insertion.
- 19 6. A system as claimed in claim 5, wherein certain said data fields being  
20 generated automatically by said system.
- 21 7. A system as claimed in claim 5, wherein certain said data fields being supplied  
22 by a user.
- 23 8. A system as claimed in claim 1, wherein at least one instance of said insertion  
24 being given an associated insertion header, wherein said insertion header includes data

- 1 fields related to the destination of said content, wherein at least one of said data fields  
2 being used to associate said insertion with at least one instance of said content.
- 3 9. A system as claimed in claim 8, wherein certain said data fields being  
4 generated automatically by said system.
- 5 10. A system as claimed in claim 8, wherein certain said data fields being supplied  
6 by a user.
- 7 11. A system as claimed in claim 1, wherein at least one instance of said story  
8 being given an associated story header, wherein said story header includes data fields  
9 related to the content and insertions associated with said story.
- 10 12. A system as claimed in claim 11, wherein certain said data fields being  
11 generated automatically by said system.
- 12 13. A system as claimed in claim 11, wherein certain said data fields being  
13 supplied by a user.
- 14 14. A system as claimed in claim 1, further comprising a user interface to permit a  
15 user to create and/or edit said story, insertion and/or content.
- 16 15. A system as claimed in claim 1, wherein at least one new instance of said  
17 content being given an insertion to define at least the destination of said new instance  
18 of said content.
- 19 16. A system as claimed in claim 1, wherein said destination includes publication,  
20 publication medium, publication zone, publication date and/or publication edition for  
21 said content.
- 22 17. A system as claimed in claim 1, wherein said central database permitting said  
23 content to be edited and defined in another insertion.
- 24 18. A system as claimed in claim 1, wherein said central database defining a link  
25 between at least two instances of said content between separate insertions, and

1 wherein an editing change in one instance of said linked content being automatically  
2 updated in all linked content.

3 19. A system as claimed in claim 1, wherein said central database defining a link  
4 between certain insertions and related content and being adapted to globally update  
5 changes in a linked insertion to all other linked insertions.

6 20. A system as claimed in claim 4, wherein a story, content or insertion created  
7 by a user of said system being checked against said *a priori* database to determine if  
8 any preset rules exist for said story, insertion and/or content.

9 21. A system as claimed in claim 4, wherein said preset data includes data related  
10 to users of said system, groups of said users of said system, default parameters of  
11 publishable events, and/or restrictions placed upon said users, said groups, and /or  
12 said publishable events.

13 22. A system as claimed in claim 4, wherein said preset data includes data  
14 defining the layout of publishable events and/or common pages between publishable  
15 events.

16 23. A system as claimed in claim 1, further comprising a communications  
17 manager to permit external content data to be updated into the central database from  
18 an external source.

19 23. A system as claimed in claim 1, further comprising an output manager being  
20 adapted to gather one or more stories, insertions and content for a given publication,  
21 publication medium, publication zone, publication date and/or publication edition and  
22 paginate said content for said publication, publication medium, publication zone,  
23 publication date and/or publication edition based on user-defined and/or preset  
24 pagination criteria, and being adapted to output said content for said publication,

1 publication medium, publication zone, publication date and/or publication edition in a  
2 preset format.

3 24. A system as claimed in claim 1, further comprising a pagination interface to  
4 permit a user to paginate said content and defining said pagination data in said content  
5 and said insertion for that content.

6 25. A system as claimed in claim 1, wherein each said insertion comprises a  
7 plurality of different content.

8 26. A system as claimed in claim 1, wherein said central database includes  
9 database translators to permit a user to translate said story, insertion and content into a  
10 preset database language.

11 27. A system as claimed in claim 1, said system being adapted to permit a user to  
12 monitor one or more said stories, content and insertion from creation to said  
13 destination, and permitting each said story, insertion and/or content to be viewed and  
14 modified from creation to said destination.

15 28. A data management and editorial system for the production and management  
16 of publishable events, comprising: a central database for storing at least one  
17 publishable story, said story comprising data content and at least one insertion related  
18 to said data content, said insertion including one or more attributes related to said data  
19 content wherein at least one attribute defines a destination for said content, wherein a  
20 single copy of said content being defined by separate insertions to appear in a plurality  
21 of separate publishable events, wherein said database linking at least two said  
22 instances of said single copy of said content and updating changes in said content to  
23 all linked instances of said content.

- 1 29. A system as claimed in claim 28, wherein said publishable event includes  
2 those instances where content is to be placed into a defined publication, publication  
3 medium, publication zone, publication date and/or publication edition.
- 4 30. A system as claimed in claim 28, wherein said publishable event includes an  
5 instance where content is destined for a print medium, broadcast medium or Internet  
6 medium.
- 7 31. A system as claimed in claim 28, wherein said content includes text, graphics,  
8 video and/or audio data that is computer readable/transmittable.
- 9 32. A system as claimed in claim 28, wherein said system further comprising an *a*  
10 *priori* database which is used to define preset data related to said story, insertion,  
11 and/or content.
- 12 33. A system as claimed in claim 28, wherein at least one instance of said content  
13 being given an associated content header, wherein said header includes data fields  
14 related to the format of said content, wherein at least one of said data fields being used  
15 to associate said content with at least one said insertion.
- 16 34. A system as claimed in claim 33, wherein certain said data fields being  
17 generated automatically by said system.
- 18 35. A system as claimed in claim 33, wherein certain said data fields being  
19 supplied by a user.
- 20 36. A system as claimed in claim 28, wherein at least one instance of said  
21 insertion being given an associated insertion header, wherein said insertion header  
22 includes data fields related to the destination of said content, wherein at least one of  
23 said data fields being used to associate said insertion with at least one instance of said  
24 content.

- 1 37. A system as claimed in claim 36, wherein certain said data fields being  
2 generated automatically by said system.
- 3 38. A system as claimed in claim 36, wherein certain said data fields being  
4 supplied by a user.
- 5 39. A system as claimed in claim 28, wherein at least one instance of said story  
6 being given an associated story header, wherein said story header includes data fields  
7 related to the content and insertions associated with said story.
- 8 40. A system as claimed in claim 39, wherein certain said data fields being  
9 generated automatically by said system.
- 10 41. A system as claimed in claim 39, wherein certain said data fields being  
11 supplied by a user.
- 12 42. A system as claimed in claim 28, further comprising a user interface to permit  
13 a user to create and/or edit said story, insertion and/or content.
- 14 43. A system as claimed in claim 32, wherein at least one story, content or  
15 insertion created by a user of said system being compared to said *a priori* database to  
16 determine if any preset rules exist for said story, insertion and/or content.
- 17 44. A system as claimed in claim 32, wherein said preset data includes data related  
18 to users of said system, groups of said users of said system, default parameters of  
19 publishable events, and/or restrictions placed upon said users, said groups, and /or  
20 said publishable events.
- 21 45. A system as claimed in claim 32, wherein said preset data includes data  
22 defining the layout of publishable events and/or common pages between publishable  
23 events.



- 1 46. A system as claimed in claim 28, further comprising a communications  
2 manager to permit external content data to be input into the central database from an  
3 external source.
- 4 47. A system as claimed in claim 28, further comprising an output manager being  
5 adapted to gather one or more stories, insertions and content for a given publication,  
6 publication medium, publication zone, publication date and/or publication edition and  
7 paginate said content for said publication, publication medium, publication zone,  
8 publication date and/or publication edition based on user-defined and/or preset  
9 pagination criteria, and being adapted to output said content for said publication,  
10 publication medium, publication zone, publication date and/or publication edition in a  
11 preset format.
- 12 48. A system as claimed in claim 28, further comprising a pagination interface to  
13 permit a user to paginate said content and defining said pagination data in said content  
14 and said insertion for that content.
- 15 49. A system as claimed in claim 28, wherein each said insertion comprises a  
16 plurality of different content.
- 17 50. A system as claimed in claim 28, wherein said central database includes a  
18 database translator to permit a user to translate said story, insertion and content into a  
19 preset database language.
- 20 51. A system as claimed in claim 28, said system being adapted to permit a user to  
21 monitor one or more said stories, content and insertion from creation to said  
22 destination, and permitting said story, insertion and/or component to be viewed and  
23 modified from creation to said destination.
- 24 52. An data management and editorial system for the production and management  
25 of publishable events, comprising: a central database for storing at least one

1 publishable story, said story comprising data content and a plurality of insertions  
2 related to said content, each said insertions including one or more attributes related to  
3 said data content wherein at least one attribute defines a destination for said content,  
4 said central database also for storing at least one common page data which defines  
5 content that is to appear substantially the same on a given page in multiple  
6 destinations, wherein all said insertions that define a destination that is part of a  
7 common page being linked together and updating changes in one linked insertion with  
8 other linked insertions.

9 53. A system as claimed in claim 52, wherein said publishable event includes  
10 those instances where content is to be placed into a publication, publication medium,  
11 publication zone, publication date and/or publication edition.

12 54. A system as claimed in claim 52, wherein said publishable event includes an  
13 instance where content is destined for a print medium, broadcast medium or Internet  
14 medium.

15 55. A system as claimed in claim 52, wherein said wherein said content includes  
16 text, graphics, video and/or audio data that is computer readable/transmittable.

17 56. A system as claimed in claim 52, wherein said central database further  
18 comprising an *a priori* database which is used to define preset data related to said  
19 story, insertion, and/or content.

20 57. A system as claimed in claim 52, wherein at least one instance of said content  
21 being given an associated content header, wherein said header includes data fields  
22 related to the format of said content, wherein at least one of said data fields being used  
23 to associate said content with at least one said insertion.

24 58. A system as claimed in claim 57, wherein certain said data fields being  
25 generated automatically by said system.

- 1 59. A system as claimed in claim 57, wherein certain said data fields being  
2 supplied by a user.
- 3 60. A system as claimed in claim 52, wherein at least one instance of said  
4 insertion being given an associated insertion header, wherein said insertion header  
5 includes data fields related to the destination of said content, wherein at least one of  
6 said data fields being used to associate said insertion with at least one instance of said  
7 content.
- 8 61. A system as claimed in claim 60, wherein certain said data fields being  
9 generated automatically by said system.
- 10 62. A system as claimed in claim 60, wherein certain said data fields being  
11 supplied by a user.
- 12 63. A system as claimed in claim 52, wherein at least one instance of said story  
13 being given an associated story header, wherein said story header includes data fields  
14 related to the content and insertions associated with said story.
- 15 64. A system as claimed in claim 63, wherein certain said data fields being  
16 generated automatically by said system.
- 17 65. A system as claimed in claim 63, wherein certain said data fields being  
18 supplied by a user.
- 19 66. A system as claimed in claim 52, further comprising a user interface to permit  
20 a user to create and/or edit said story, insertion and/or content.
- 21 67. A system as claimed in claim 56, wherein a story, content or insertion created  
22 by a user of said system being compared to said *a priori* database to determine if any  
23 preset rules exist for said story, insertion and/or content.
- 24 68. A system as claimed in claim 67, wherein said preset data includes data related  
25 to users of said system, groups of said users of said system, default parameters of

- 1 publishable events, and/or restrictions placed upon said users, said groups, and /or  
2 said publishable events.
- 3 69. A system as claimed in claim 67, wherein said preset rules includes data  
4 defining the layout of publishable events and/or common pages between publishable  
5 events.
- 6 70. A system as claimed in claim 52, further comprising a communications  
7 manager to permit external content data to be input into the central database from an  
8 external source.
- 9 71. A system as claimed in claim 52, further comprising an output manager being  
10 adapted to gather one or more stories, insertions and content for a given publication,  
11 publication medium, publication zone, publication date and/or publication edition and  
12 paginate said content for said publication, publication medium, publication zone,  
13 publication date and/or publication edition based on user-defined and/or preset  
14 pagination criteria, and being adapted to output said content for said publication,  
15 publication medium, publication zone, publication date and/or publication edition in a  
16 preset format.
- 17 72. A system as claimed in claim 52, further comprising a pagination interface to  
18 permit a user to paginate said content and defining said pagination data in said content  
19 and said insertion for that content.
- 20 73. A system as claimed in claim 52, wherein at least one said linked insertion  
21 comprises a plurality of different content.
- 22 74. A system as claimed in claim 52, wherein said central database includes a  
23 database translator to permit a user to translate said story, insertion and content into a  
24 preset database language.

1 75. A system as claimed in claim 52, said system being adapted to permit a user to  
2 monitor one or more said stories, content and insertion from creation to said  
3 destination, and permitting said story, insertion and/or content to be viewed and  
4 modified from creation to said destination.

5 76. A method of creating a publishable story, comprising the steps of:  
6 creating one or more content items related to a publishable event;  
7 creating at least one insertion for said content items, said insertion including  
8 attribute data including destination data to direct said content to a specified  
9 destination;  
10 associating two or more said content items and two or more said insertions  
11 related to said publishable story to form a unified data structure; and  
12 storing said content items, insertions and associations on a database.

13 77. A method as claimed in claim 76, further comprising the steps of linking one  
14 or more of said content items and creating additional insertions for each said linked  
15 content item, wherein additional instances of a linked content are provided in said  
16 additional insertions and wherein a change in one linked content item is automatically  
17 applied to all linked content items.

18 78. A method as claimed in claim 76, further comprising the steps of determining  
19 if common page rules exist and linking two or more insertions whose content appears  
20 on common pages.

21 79. A method as claimed in claim 76, further comprising the steps of establishing  
22 an *a priori* database of preset rules and predefined data and comparing one or more  
23 instances of said content and insertions to said *a priori* database to determine if said  
24 content or said insertions violates said preset rules or predefined data.

1 80. A method as claimed in claim 76, further comprising the steps of grouping two  
2 or more insertions and content items destined for a given publication, publication  
3 medium, publication zone, publication date, or publication edition and outputting said  
4 grouping to said destination.

5 81. A method as claimed in claim 76, further comprising the steps of paginating  
6 said content for a specific publication, publication medium, publication zone,  
7 publication date, or publication edition.

8 82. A system as claimed in claim 28, wherein said destination includes  
9 publication, publication medium, publication zone, publication date and/or  
10 publication edition for said content.

11 83. A system as claimed in claim 28, wherein said central database permitting said  
12 content to be edited and defined in another insertion.

13 84. A system as claimed in claim 52, wherein said destination includes  
14 publication, publication medium, publication zone, publication date and/or  
15 publication edition for said content.

16 85. A system as claimed in claim 52, wherein said central database permitting said  
17 content to be edited and defined in another insertion.

18 86. A system as claimed in claim 52, wherein said central database defining a link  
19 between like content between separate insertions, and wherein an editing change in  
20 one instance of said linked content being automatically updated in linked content.

21 87. A system as claimed in claim 28, wherein said central database defining a link  
22 between certain insertions and related content and being adapted to globally update  
23 changes in a linked insertion to other linked insertions.

24 88. A method of creating a publishable story, comprising the steps of:  
25 creating one or more content items related to a publishable event;

1           creating at least one insertion for said content items, said insertion including  
2           attribute data including destination data to direct said content to a specified  
3           destination;  
4           creating separate instances of a single copy of said content item and linking at  
5           least two instances of said single copy of said content item defined by said insertions;  
6           updating changes in said single copy of said content item to other linked  
7           instances of said content item;  
8           associating two or more said content items and two or more said insertions  
9           related to said publishable story to form a unified data structure; and  
10          storing said content items, insertions and associations on a database.

11   89.    A method as claimed in claim 88, further comprising the steps of determining  
12   if common page rules exist and linking two or more insertions whose content appears  
13   on common pages.

14   91.    A method as claimed in claim 88, further comprising the steps of establishing  
15   an *a priori* database of preset rules and predefined data and comparing said content  
16   and insertions to said *a priori* database to determine if said content or said insertions  
17   violates said preset rules or predefined data.

18   92.    A method as claimed in claim 88, further comprising the steps of grouping two  
19   or more insertions and content items destined for a given publication, publication  
20   medium, publication zone, publication date, or publication edition and outputting said  
21   grouping to said destination.

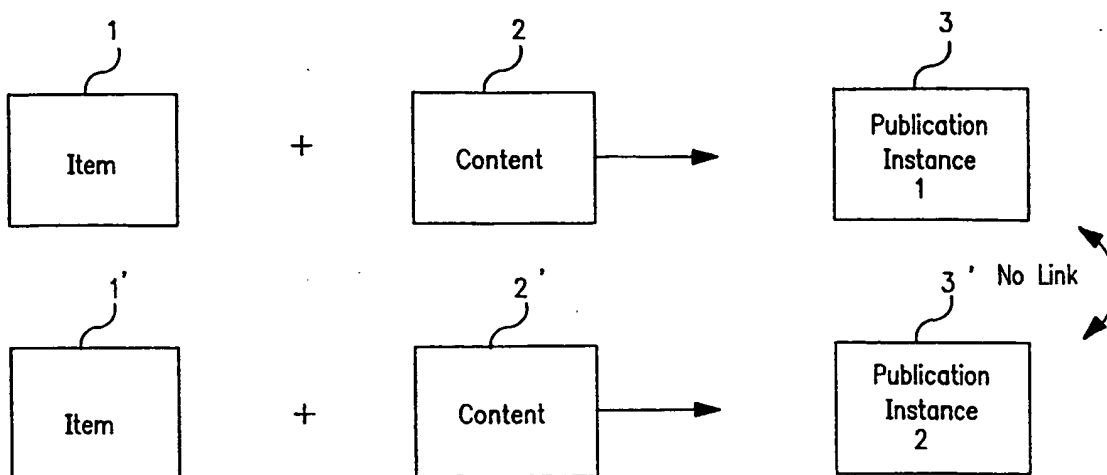
22   93.    A method as claimed in claim 88, further comprising the steps of paginating  
23   said content for a specific publication, publication medium, publication zone,  
24   publication date, or publication edition.

25   94.    A method of creating a publishable story, comprising the steps of:

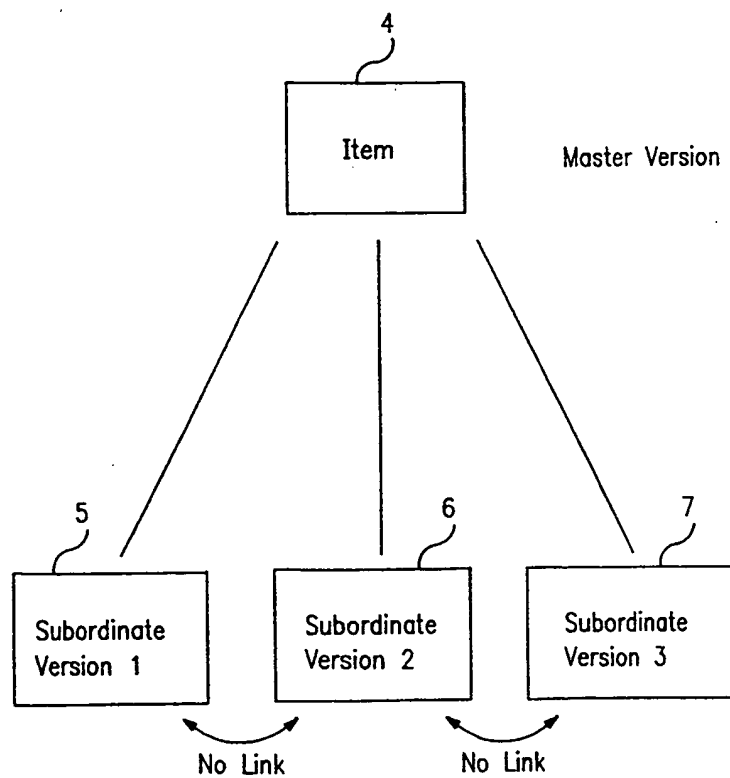
- 1           creating one or more content items related to a publishable event;
- 2           creating at least one insertion for said content items, said insertion including
- 3           attribute data including destination data to direct said content to a specified
- 4           destination;
- 5           determining if common page rules exist and linking insertions whose content
- 6           appears on common pages.
- 7           associating two or more said content items and two or more said insertions
- 8           related to said publishable story to form a single, unified data structure; and
- 9           storing said content items, insertions and associations on a database.
- 10       95.    A method as claimed in claim 94, further comprising the steps of linking one
- 11       or more of said content items and creating additional insertions for each said linked
- 12       content item, wherein additional instances of a linked content are provided in said
- 13       additional insertions and wherein a change in one linked content item is automatically
- 14       applied to linked content items.
- 15       96.    A method as claimed in claim 94, further comprising the steps of establishing
- 16       an *a priori* database of preset rules and predefined data and comparing said content
- 17       and insertions to said *a priori* database to determine if said content or said insertions
- 18       violates said preset rules or predefined data.
- 19       97.    A method as claimed in claim 94, further comprising the steps of grouping two
- 20       or more insertions and content items destined for a given publication, publication
- 21       medium, publication zone, publication date, or publication edition and outputting said
- 22       grouping to said destination.
- 23       98.    A method as claimed in claim 94, further comprising the steps of paginating
- 24       said content for a specific publication, publication medium, publication zone,
- 25       publication date, or publication edition.



1 / 10



**FIG. 1**  
Prior Art



**FIG. 2**  
Prior Art

**SUBSTITUTE SHEET (RULE 26)**



3 / 10

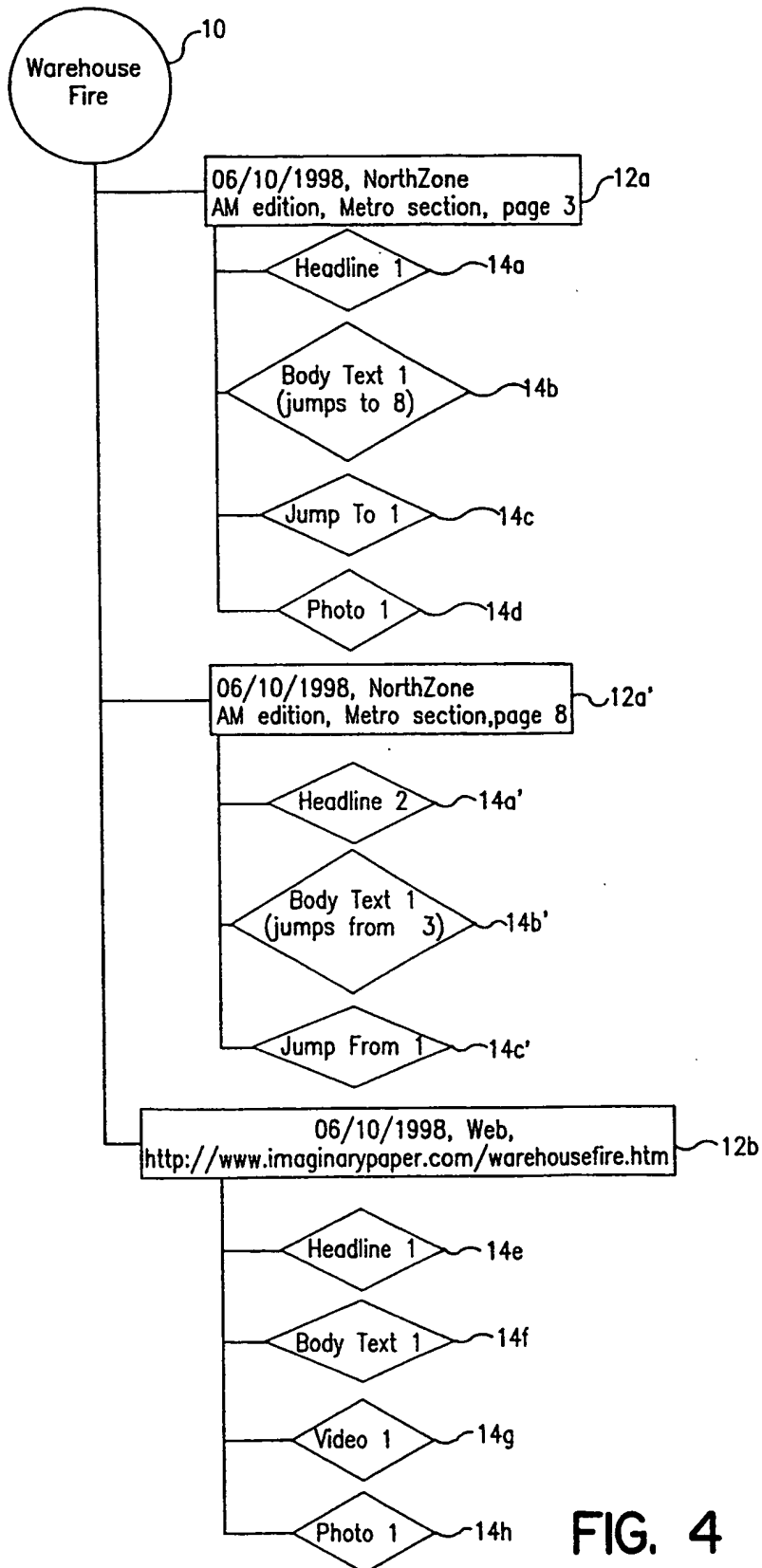
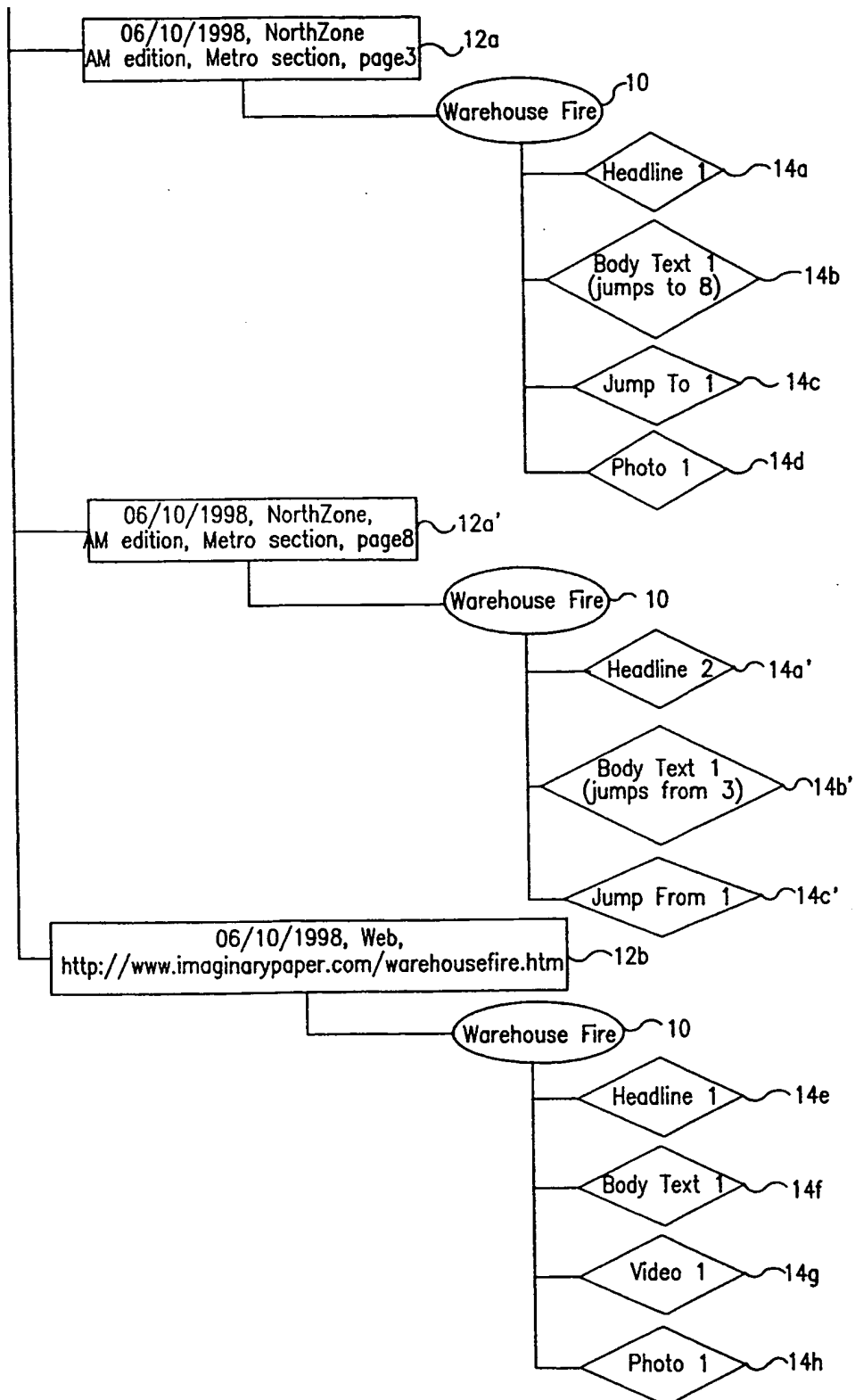


FIG. 4

SUBSTITUTE SHEET (RULE 26)

4 / 10



5 / 10

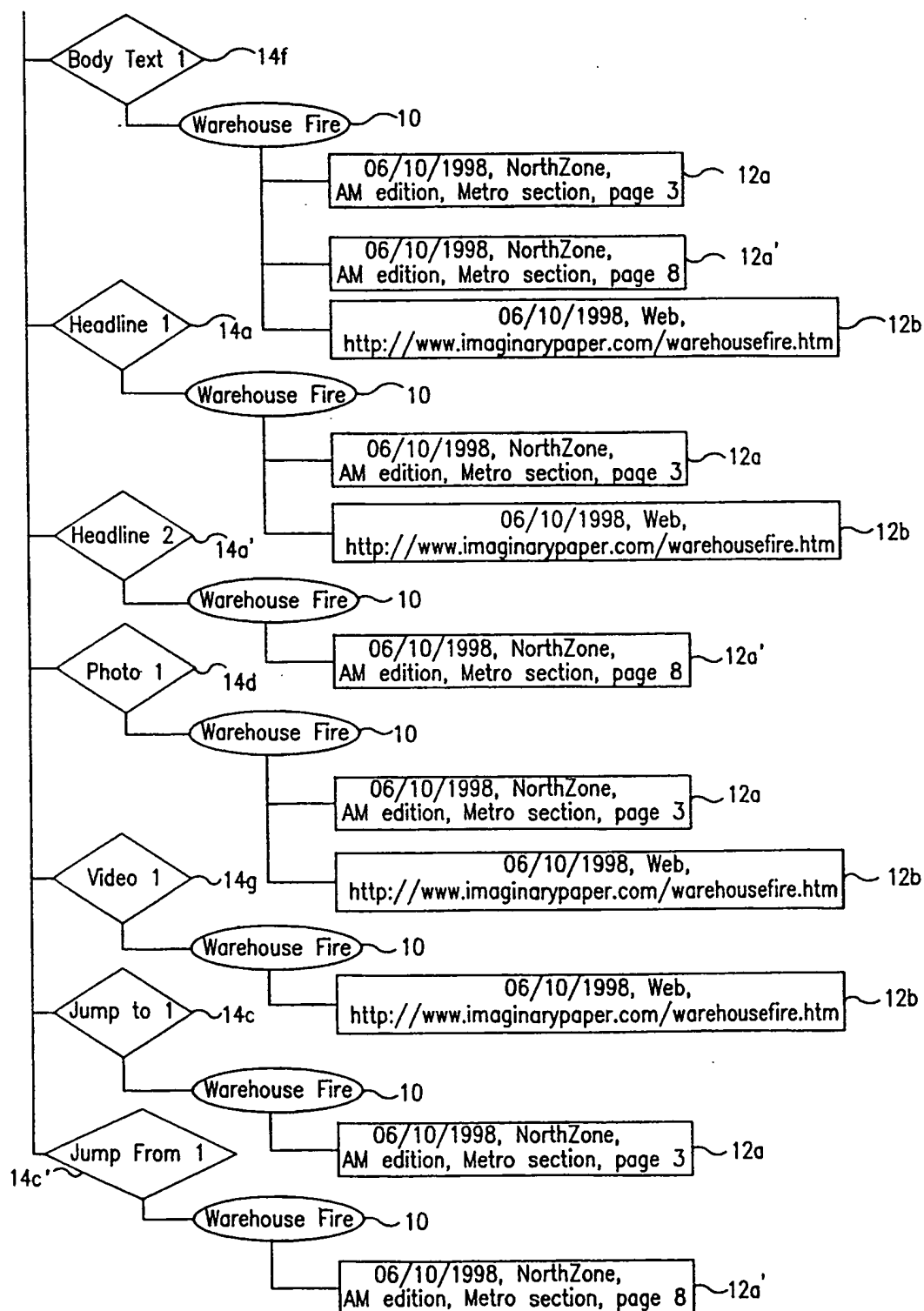


FIG. 6

SUBSTITUTE SHEET (RULE 26)

6/10

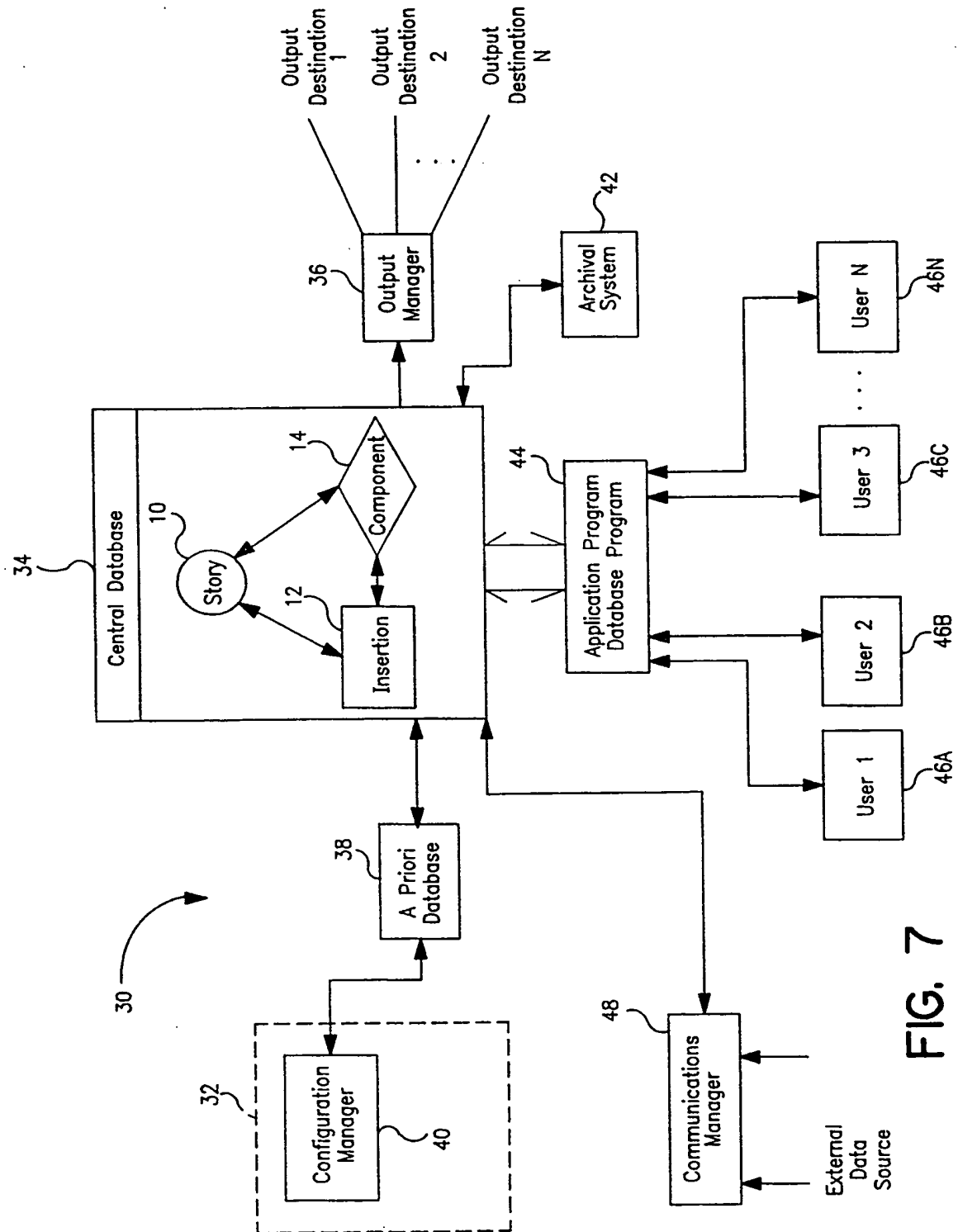


FIG. 7

7/ 10

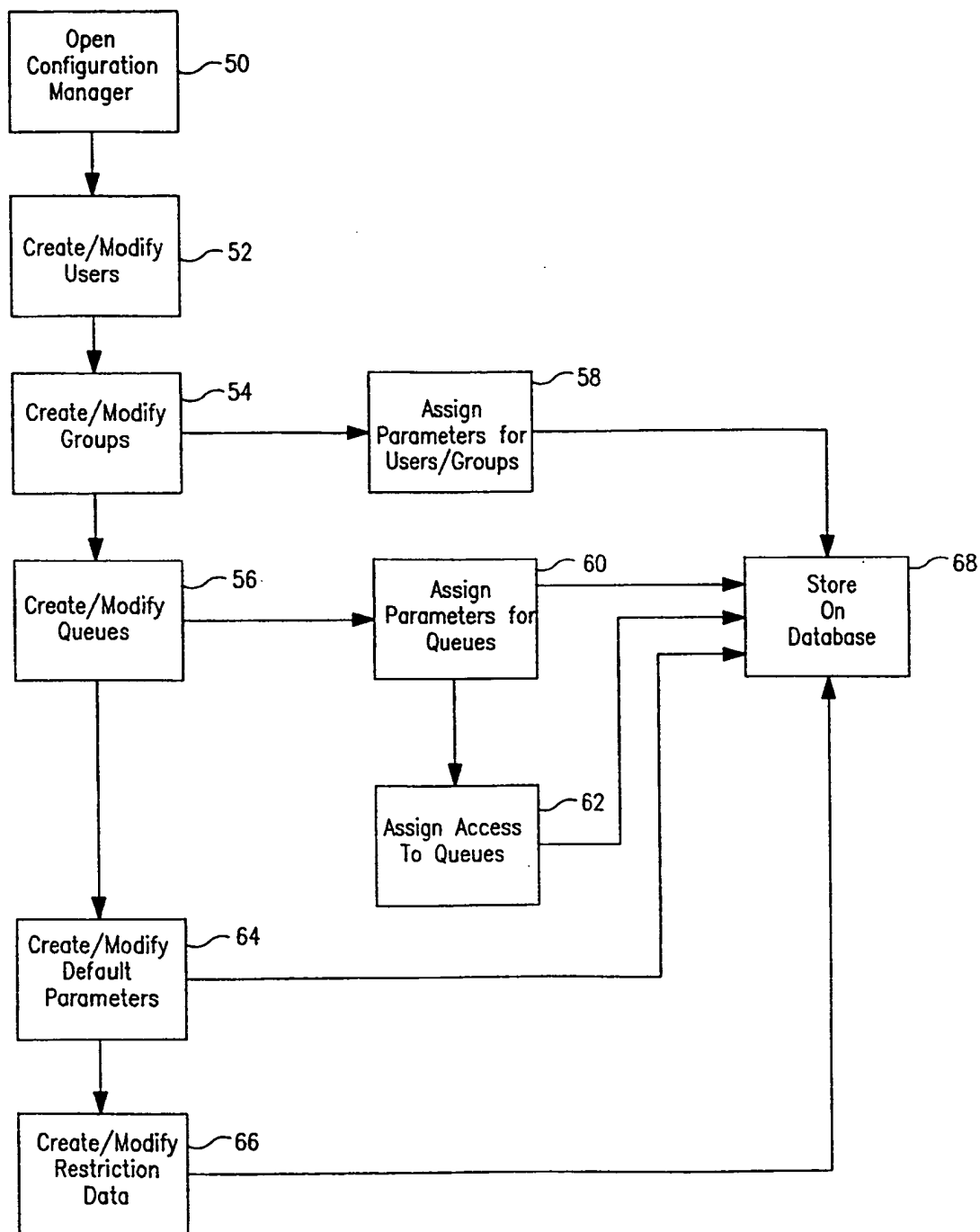


FIG. 8A

8 / 10

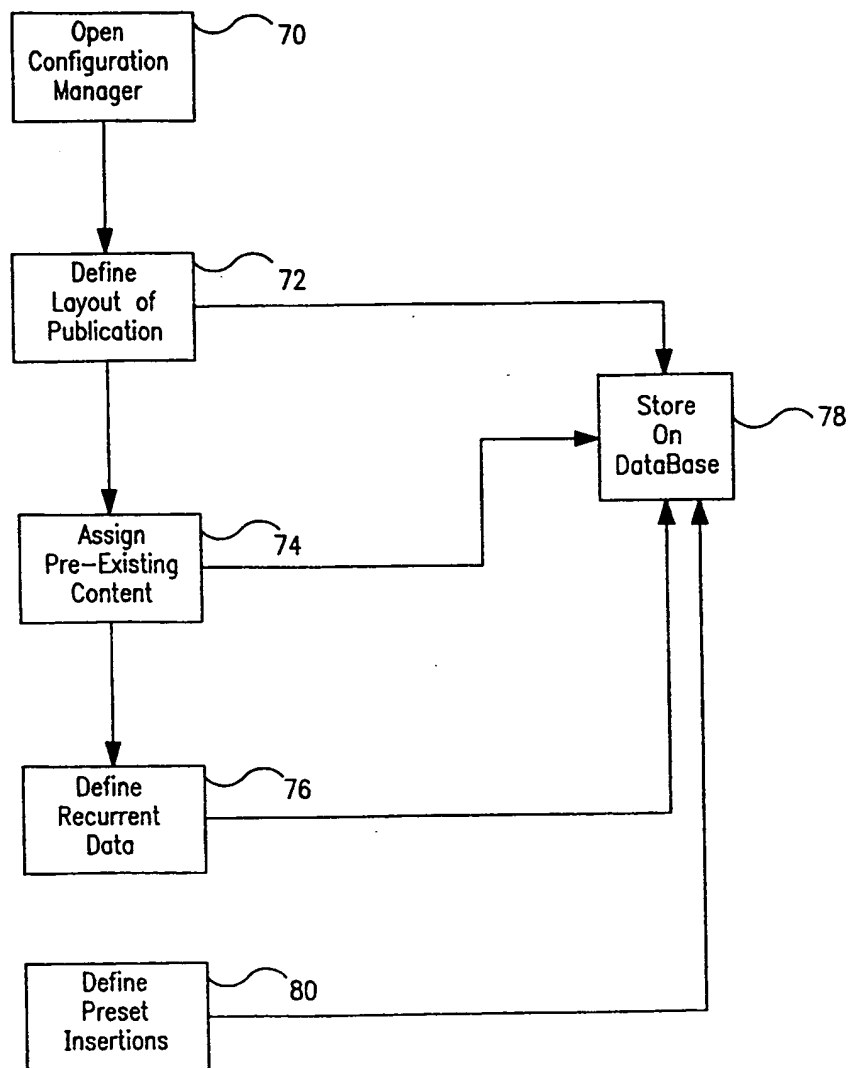


FIG. 8B



9/10

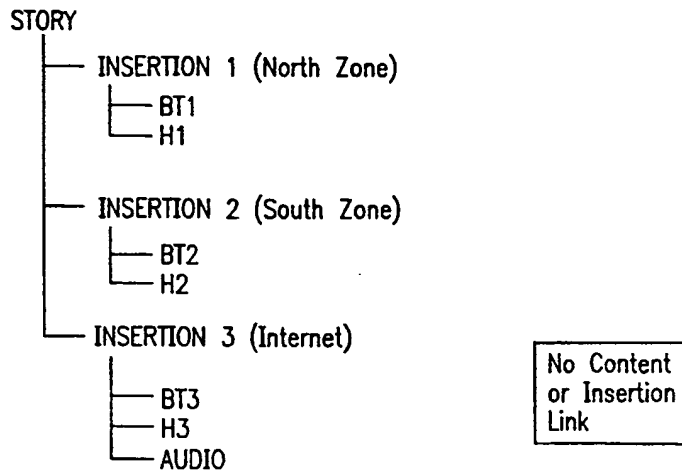


FIG. 9

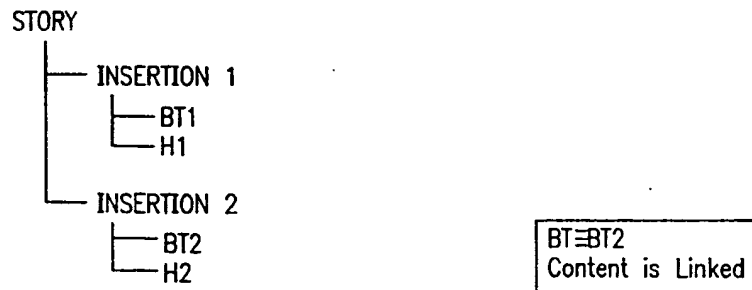


FIG. 10



FIG. 11

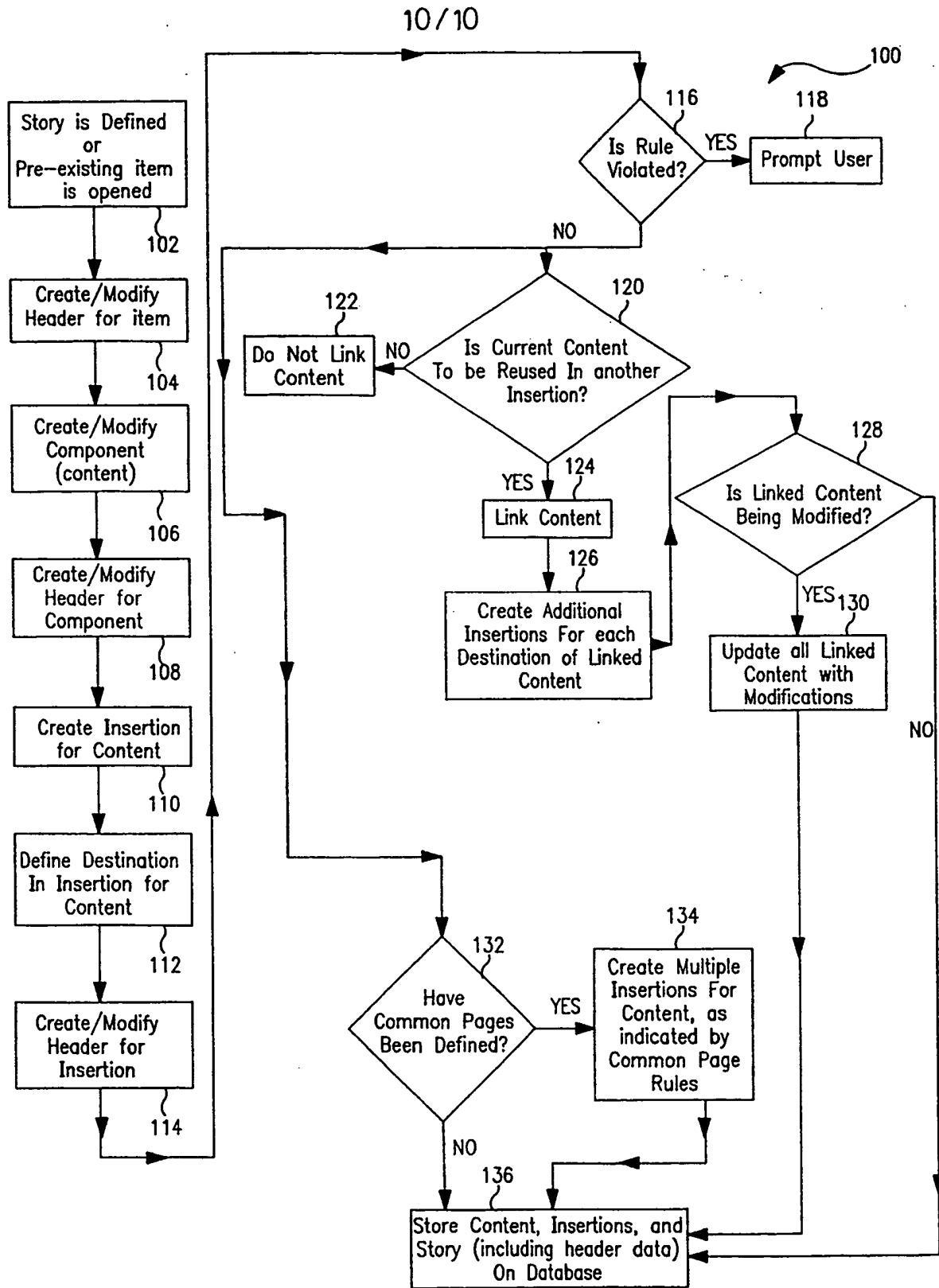


FIG. 12  
SUBSTITUTE SHEET (RULE 26)

# INTERNATIONAL SEARCH REPORT

International Application No

PC./US 99/13633

**A. CLASSIFICATION OF SUBJECT MATTER**  
IPC 6 G06F17/21 G06F17/60

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>WO 98 10356 A (DESIGN INTELLIGENCE INC) 12 March 1998 (1998-03-12)</p> <p>abstract; figures 2-6 page 1, line 10 - line 20 page 2, line 25 - line 32 page 3, line 10 - line 16 page 10, line 18 -page 11, line 2 page 17, line 3 -page 18, line 17 -/--</p>	<p>1-17, 21-25, 28-42, 44,45, 47-49, 52-66, 68,69, 71-73, 76, 80-85, 88, 92-94, 97,98</p>

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

\* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

7 October 1999

Date of mailing of the international search report

13/10/1999

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
Fax: (+31-70) 340-3016

Authorized officer

van der Weiden, A

## INTERNATIONAL SEARCH REPORT

International Application No

PC1/US 99/13633

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>-----</p> <p>US 5 181 162 A (BOER JAN H ET AL)  19 January 1993 (1993-01-19)  cited in the application  abstract  column 2, line 30 -column 4, line 13  column 9, line 25 -column 10, line 5</p> <p>-----</p>	1,28,52, 76,88,94
A	<p>GB 2 302 524 A (CYBERGRAPHIC SYST PTY LTD)  22 January 1997 (1997-01-22)  abstract  page 1, line 8 - line 9; claim 12</p> <p>-----</p>	1,28,52, 76,88,94
A	<p>ANONYMOUS: "Multi-Environment  Preprocessor for Documentation Publishing  System"  IBM TECHNICAL DISCLOSURE BULLETIN,  vol. 37, no. 4B, pages 85-88, XP000451181  New York, US  paragraph '0003! - paragraph '0004!;  figure 2</p> <p>-----</p>	1,28,52, 76,88,94
P,A	<p>US 5 892 513 A (FAY ROBERT G)  6 April 1999 (1999-04-06)  abstract; claims 1,2; figures 2,4</p> <p>-----</p>	1,28,52, 76,88,94
P,A	<p>HOFSTETTER I: "Multimedia applications  for local newspapers and local  information"  COMPUTER NETWORKS AND ISDN SYSTEMS,  vol. 30, no. 13,  3 August 1998 (1998-08-03), page 1223-1232  XP004147407  ISSN: 0169-7552  page 1223 -page 1227</p> <p>-----</p>	1,28,52, 76,88,94

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No

PCT/US 99/13633

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
WO 9810356	A	12-03-1998	US 5895476 A	20-04-1999
			US 5903902 A	11-05-1999
			US 5895477 A	20-04-1999
			AU 4334197 A	26-03-1998
			EP 0925542 A	30-06-1999
US 5181162	A	19-01-1993	EP 0462242 A	27-12-1991
			WO 9108538 A	13-06-1991
GB 2302524	A	22-01-1997	DE 19653435 A	25-06-1998
			FR 2757655 A	26-06-1998
			AU 5597996 A	09-01-1998
US 5892513	A	06-04-1999	NONE	